

UNISYS

V Series

**MCP/VS System
Software**

**Operations
Guide**

Volume 2: System Commands

Relative to Release
Level 2.0

Priced Item

July 1987
Distribution Code SD
Printed in U S America
5026248

UNISYS

V Series

**MCP/VS System
Software**

**Operations
Guide**

Volume 2: System Commands

Copyright © 1987 Unisys Corporation
All Rights Reserved
Unisys is a trademark of Unisys Corporation.

Relative to Release
Level 2.0

Priced Item

July 1987
Distribution Code SD
Printed in U S America
5026248

The names, places, and/or events used in this publication are not intended to correspond to any individual, group, or association existing, living, or otherwise. Any similarity or likeness of the names, places, and/or events with the names of any individual living or otherwise, or that of any group or association is purely coincidental and unintentional.

NO WARRANTIES OF ANY NATURE ARE EXTENDED BY THE DOCUMENT. Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed Program Product License or Agreement to purchase or lease equipment. The only warranties made by Unisys, if any, with respect to the products described in this document are set forth in such License or Agreement. Unisys cannot accept any financial or other responsibility that may be the result of your use of the information in this document or software material, including direct, indirect, special or consequential damages.

You should be very careful to ensure that the use of this information and/or software material complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

Comments or suggestions regarding this document should be submitted on a Field Communication Form (FCF) with the CLASS specified as 2 (S.W.:System Software), and the Type specified as 1 (F.T.R.), and the product specified as the 7-digit form number of the manual (for example, 5026248).



Publication Change Notice (PCN)

UNISYS

Date

May 1988

Form — PCN Number

5026248-001

Title

V Series MCP/VS Systems Software Operations Guide Volume 2: System Commands

Description

This PCN updates the document to reflect the 2.1 release (May, 1988).

Replace these pages

v through xvii-A

2-5 through 2-7

2-11

2-15

2-39 through 2-41

2-107 through 2-109

2-173 through 2-181

2-193

2-205

2-219

2-233 through 2-235

2-243

2-247

2-287 through 2-289

2-293

2-315 through 2-317

2-349

2-359 through 2-361

Add these pages

—

2-8-A

2-12-A

2-16-A

—

—

—

2-193-B

2-208-A

2-219-B

—

—

—

—

—

—

2-328-A

2-350-A

2-361-B

Copyright © 1988 Unisys Corporation

Distribution Code SD

Printed in the United States

AFFECTED PAGES

Page	Issue
Title	Original
v through xvii-A	PCN-001
xviii	Blank
xix through xxi	Original
xxii	Blank
1-1 through 1-14	Original
2-1 through 2-4	Original
2-5 through 2-8-B	PNC-001
2-9 through 2-10	Original
2-11 through 2-12-B	PCN-001
2-13 through 2-14	Original
2-15 through 2-16-B	PCN-001
2-17 through 2-38	Original
2-39 through 2-42	PCN-001
2-43 through 2-106	Original
2-107 through 2-110	PCN-001
2-111 through 2-172	Original
2-173 through 2-182	PCN-001
2-183 through 2-192	Original
2-193 through 2-194	PCN-001
2-195 through 2-204	Original
2-205 through 2-208-B	PCN-001
2-209 through 2-118	Original
2-219 through 2-220	PCN-001
2-221 through 2-232	Original
2-233 through 2-236	PCN-001
2-237 through 2-242	Original
2-243 through 2-244	PCN-001
2-245 through 2-246	Original
2-247 through 2-248	PCN-001
2-249 through 2-286	Original
2-287 through 290	PCN-001
2-291 through 2-292	Original
2-293 through 2-294	PCN-001
2-295 through 2-314	Original
2-315 through 2-318	PCN-001
2-319 through 2-328	Original
2-328-A through 2-328-B	PCN-001
2-329 through 2-348	Original
2-349 through 2-350-B	PCN-001
2-351 through 2-358	Original
2-359 through 2-362	PCN-001
2-363 through 2-366	Original
A-1 through A-6	Original
B-1 through B-9	Original
B-10	Blank
Glossary-1 through Glossary-23	Original
Glossary-24	Blank
1 through 14	Original

AFFECTED PAGES

Page	Issue	Page	Issue
Title	Original	2-331 through 2-332	PCN-002
v through xiv	PCN-001	2-333 through 2-348	Original
xv through xvi	PCN-002	2-349 through 2-350-B	PCN-001
xvii through xvii-A	PCN-001	2-351 through 2-358	Original
xviii	Blank	2-359 through 2-361-A	PCN-001
xix through xxi	Original	2-361-B through 2-362	PCN-002
xxii	Blank	2-363 through 2-366	Original
1-1 through 1-14	Original	A-1 through A-6	Original
2-1 through 2-4	PCN-002	B-1 through B-9	Original
2-5 through 2-8-B	PCN-001	B-10	Blank
2-9 through 2-10	Original	glossary-1 through glossary-4	Original
2-11 through 2-12-B	PCN-001	glossary-5 through glossary-10	PCN-002
2-13 through 2-14	Original	glossary-11 through glossary-12	Original
2-15 through 2-16-B	PCN-001	glossary-13 through glossary-16	PCN-002
2-17 through 2-20	PCN-002	glossary-17 through glossary-18	Original
2-21 through 2-38	Original	glossary-19 through glossary-20	PCN-002
2-39 through 2-42	PCN-001	glossary-21 through glossary-22	Original
2-43 through 2-68	Original	glossary-23	PCN-002
2-69 through 2-72	PCN-002	glossary-24	Blank
2-73 through 2-104	Original	1 through 14	Original
2-105 through 2-108	PCN-002		
2-109 through 2-110	PCN-001		
2-111 through 2-212-B	PCN-002		
2-213 through 2-226	Original		
2-227 through 2-230	PCN-002		
2-231 through 2-252	Original		
2-253 through 2-254-B	PCN-002		
2-255 through 2-172	Original		
2-173 through 2-182	PCN-001		
2-183 through 2-192	Original		
2-193 through 2-194	PCN-001		
2-195 through 2-204	Original		
2-205 through 2-208-B	PCN-001		
2-209 through 2-118	Original		
2-219 through 2-220	PCN-001		
2-221 through 2-232	Original		
2-233 through 2-236	PCN-001		
2-237 through 2-242	Original		
2-243 through 2-244	PCN-001		
2-245 through 2-246	Original		
2-247 through 2-248	PCN-001		
2-249 through 2-272	Original		
2-273 through 2-274	PCN-002		
2-275 through 2-286	Original		
2-287 through 288	PCN-001		
2-289 through 2-290	PCN-002		
2-291 through 2-292	Original		
2-293 through 2-296	PCN-002		
2-297 through 2-314	Original		
2-315 through 2-318	PCN-001		
2-319 through 2-326	Original		
2-327 through 2-328	PCN-002		
2-328-A through 2-328-B	PCN-001		
2-329 through 2-330	Original		



TABLE OF CONTENTS

Section	Title	Page
	ABOUT THIS MANUAL	xix
	Scope And Purpose	xix
	Prerequisites	xix
	Organization Of This Manual	xix
	Related Product Information	xx
1	INTRODUCTION TO THE SYSTEM COMMANDS	1-1
	System Command Overview	1-1
	Keyboard Input Messages vs. MCP Control Instructions	1-2
	Entering Commands Through The ODT	1-3
	MCP Response To ODT Input	1-4
	Entering Commands Through a Programmatic ZIP or ZIPSPO	1-5
	Entering Commands Through A ZIPSPO	1-5
	Entering Commands Through A SPOMESSAGE BCT	1-6
	Entering Commands Through Punched Cards	1-6
	Entering Commands Through CANDE, PROPS, or WFL	1-7
	Entering Commands Through RJE	1-7
	Command Details	1-8
	File-Identifiers	1-9
	Special Characters	1-11
	Command Restrictions	1-12
	Access Level	1-12
2	THE SYSTEM COMMANDS	2-1
	AC (Purge And Place Tape Number In Label)	2-2
	AD (Specify ODT/OCS Screen Parameters)	2-4
	ADD (Copy Files From Library Tape To Disk Or Pack)	2-9
	AFTER (Hold Program)	2-10
	AJ (Display Active Jobs)	2-12
	ALLOCATE (Create Disk Space And Directory Entries)	2-13
	ALTER (Change System Parameters)	2-16
	AT (Pass Message To Remote Host)	2-21
	AT (Initiate BNA Job Transfer)	2-22
	AX (Respond To An ACCEPT)	2-24
	AX (Pre-Respond To An ACCEPT)	2-25
	BCL (Mark Beginning Of Card Data File)	2-27
	BEGINUSER (Mark Beginning Of Job Stream)	2-28
	BF, BFP (List Backup Files)	2-30
	BFRF (Search, List, Remove Backup Files)	2-32
	Executing The BFRF Intrinsic	2-37
	BINARY (Read Binary Card Deck)	2-38
	BO (Blackout Access Code)	2-39
	BT (Begin TRAK)	2-40

TABLE OF CONTENTS (Continued)

Section	Title	Page
	BUFFER (Set MCS Result Queue Size)	2-42
	CA (List Port Level Candidates)	2-43
	CC (Enter Control Instructions)	2-45
	CD, CDP (List Inactive Pseudo Decks)	2-47
	CHANGE (Change File-Id)	2-49
	Pseudo-Device File-Identifiers	2-50
	CHARGE (Insert Charge Number)	2-52
	CHECK (Verify Library Tape)	2-53
	CK (Test Peripheral Device)	2-54
	CN (Display Tape Number)	2-55
	COMPARE (Compare Files On Different Media)	2-56
	COMPILE (Compile Program)	2-57
	Slash Parameters	2-61
	Getting Programs In The Correct Format	2-62
	COPY (Invoke File Transfer Utility)	2-63
	COPYADD (Transfer Files)	2-73
	CV, CVP (Convert Punch Backup File To Pseudo Card Deck)	2-74
	User-Coded LDCNTL	2-74
	DA, DAP (Deactivate Pseudo Reader)	2-76
	DATA/DATAB (Mark Beginning Of Card Data File)	2-78
	DBCA (Clear SSP Address)	2-80
	DBCL (Cancel DLP)	2-82
	DBDR (Produce SSP Content Report)	2-83
	DBIC (Initialize Controller Parameters)	2-84
	DBMM (Set/Reset Maintenance Mode)	2-85
	DBSR (Produce Block Lockout Table Report)	2-86
	DBUA (Unlock Address)	2-87
	DBUL (Clear Locked Files On Shared Systems)	2-88
	DC (Display Compilation Data)	2-89
	Displaying ICM Compilation Data	2-90
	DD (Delete DLP)	2-91
	DEBUG (Attach Program To DEBUG Session)	2-92
	DIR (List File Directory)	2-99
	DISK (Add A Disk)	2-96
	DISPLAY/WHATS (Display Information)	2-101
	DL (Delete Peripheral Devices)	2-105
	DLP (Add DLPs)	2-106
	DM (Dump And Continue)	2-110
	Memory Dumps Of A User Task	2-110
	Memory Dumps Of The Operating System	2-110
	DP (Dump And Discontinue)	2-112
	DQ (Display System Messages)	2-113
	DR (Date Reset)	2-114

TABLE OF CONTENTS (Continued)

Section	Title	Page
	DS (Discontinue Program)	2-115
	DUMP (Copy Files To Library Tape)	2-117
	DX (Display DLPs Attached To An Exchange)	2-118
	EBCDIC (Mark Beginning Of Card Data File)	2-119
	ED (Enter DEBUG Session)	2-120
	END/ENDCTL (Mark End of Card File Or Control Instructions)	2-121
	ENDAT (End Job Transfer)	2-122
	ENDUSER (Mark End Of Job Stream)	2-123
	ES (End Screen Display)	2-125
	ET (End TRAK)	2-126
	EXECUTE (Execute Library Program)	2-127
	Slash Parameters	2-130
	FI (List Files In Use By A Program)	2-131
	FILE (Equate File-Name With File-Identifier Or Medium)	2-132
	Automatic Printer Assignment	2-137
	FM (Direct File To Device -- Special Forms Response)	2-141
	FN (List File Names)	2-142
	FP (Flush Program)	2-143
	FR (Designate Final Reel)	2-144
	GENERATE (Schedule DMPALL)	2-145
	GO (Resume Stopped Program)	2-147
	GT (Initiate Trace)	2-148
	HL (Request A Halt/Load)	2-150
	HN (Identify Or List Host Name)	2-152
	ID (Invoke DEBUG Session)	2-154
	IL (Assign Labeled File)	2-155
	IN (Insert Data Into Program)	2-157
	INSERT (Insert Data Into Program)	2-159
	IR (Initiate DMSII Recovery)	2-161
	KA (Analyze Disk Directory)	2-162
	KP (Print Disk Segments)	2-165
	KS (Analyze Disk Space)	2-166
	KX (Obtain User Disk Space)	2-168
	AUTO KX	2-169
	LABEL1 (Change Tape Label Format)	2-170
	LC (Insert Log Comment)	2-174
	LD, LDP (Create Pseudo Card Deck)	2-176
	LH (Load Firmware Into A Channel Controller or DCP)	2-177
	LI (Log In)	2-179
	LN (Transfer and Print Log)	2-180
	LO (Log Out)	2-182
	LOAD (Move Files From Library Tape to Disk or Pack)	2-183
	LOCK (Lock Executing Program)	2-184

TABLE OF CONTENTS (Continued)

Section	Title	Page
	LP (Lock Program)	2-185
	MEMDUMP (Cause Abnormal EOJ Memory Dump)	2-186
	MEMORY (Allocate Memory For Program Execution)	2-187
	MERGE (Transfer Files From Multiple Sources, Resolving Duplicates)	2-189
	MOVE (Move Files To New Media)	2-190
	MR (Remove Duplicate File)	2-191
	MX (Display Jobs In Mix)	2-193
	NET (Network Control Command)	2-194
	NL (Resume At New Location)	2-197
	NT (Terminate Trace)	2-198
	NW (Networking Command Prefix)	2-199
	OF (Indicate Optional File)	2-200
	OK (Continue Processing Suspended Program)	2-201
	OL (Display Peripheral Status)	2-203
	OT (Display Program Data)	2-209
	OU (Direct File To Device)	2-210
	PA (Analyze Pack Directory)	2-212
	PACK (Add A Pack)	2-213
	PATCH (Patch File)	2-215
	Verifying Patch Records	2-218
	PB (Print Backup File)	2-220
	PC, PCP (Create Punch Backup File)	2-221
	PD (Print Directory)	2-224
	PERFORM (Schedule DMPALL)	2-226
	PG (Purge Magnetic Tape)	2-228
	PM (Print Memory Dump)	2-230
	PO (Power Off Pack)	2-239
	PP (Print Pack Sectors)	2-240
	PR (Change Priority)	2-241
	PRINT (Print Files)	2-243
	Executing PBDPRN	2-250
	Auto Printing	2-250
	PRIORITY (Set Priority)	2-252
	PS (Analyze Pack Space)	2-254
	QD (Query DEBUG Sessions)	2-255
	QT (Quit Program Operation)	2-256
	QWIKPOOL (Set Program Overlay Buffer Pool)	2-258
	QWKMEM (Set MCP Overlay Buffer Pool)	2-259
	RA (Remove After-Linkage)	2-260
	RD, RDP (Remove Pseudo Decks)	2-261
	REMOVE (Delete Files)	2-263
	RERUN (Log a Rerun Operation)	2-266
	RF, RFP (Remove Backup Files)	2-267

TABLE OF CONTENTS (Continued)

Section	Title	Page
	RM (Remove Duplicate File)	2-269
	RN, RNP (Activate Pseudo Reader)	2-271
	RO (Reset System Options)	2-273
	RP (Ready And Purge Magnetic Tape)	2-274
	RQ (Remove Storage Queue Entry)	2-276
	RS (Remove Job From Schedule)	2-277
	RUN (Initiate Execution Of Program)	2-278
	RW (Rewind And Unload Tape)	2-279
	RX (Return Deleted Space)	2-280
	RY (Ready Peripheral Device)	2-282
	SB (Interrogate Database Activity)	2-284
	SD (Set Deck Limit)	2-286
	SHOW (Display System Information)	2-287
	SI (Set Remote Device Identification)	2-291
	SK (Skip Backup Records)	2-292
	SM (Send System Message)	2-293
	SN (Purge And Place Tape Number In Label)	2-294
	SO (Set System Options)	2-296
	SP (Display/Set/Change Database Parameters)	2-300
	SPO (Simulate SPO Through Card Reader)	2-301
	SQ (Squash Disk)	2-302
	SQP (Squash Pack)	2-303
	SS (Send Message)	2-306
	ST (Suspend Program Processing)	2-307
	START (Restart A Stopped Program)	2-308
	STOP (Suspend Program Processing)	2-309
	SV (Save A Peripheral Unit)	2-310
	SW (Set Programmatic Switches)	2-311
	TEST (Mark A Test Operation)	2-313
	TI (Display Time Statistics)	2-314
	TIME (Discontinue Program After A Specified Amount Of Time)	2-315
	TL (Transfer Log)	2-316
	TM (Write Tape Mark)	2-318
	TO (Display System Options Status)	2-320
	TR (Time Reset)	2-321
	TRACE (Initiate Trace)	2-322
	UL (Assign Unlabeled File)	2-324
	UNIT (Add Peripheral Devices)	2-326
	UNLOAD (Move Library Files Between Media, Remove Old Files)	2-337
	UP (Unlock Program)	2-338
	UR (Inhibit Or Uninhibit Unit)	2-339
	USER (Assign Access Code)	2-341
	VALUE (Insert Value Into A Program)	2-343

TABLE OF CONTENTS (Continued)

Section	Title	Page
	WB (Display Bound Programs)	2-345
	WC (Display Available Memory)	2-346
	WD (Display Date)	2-347
	WJ (Display Waiting Jobs).	2-349
	WM (Display MCP Version)	2-351
	WO (Display Active User QWIK Operation Counts)	2-353
	WQ (Display STOQUE Count)	2-355
	WS (Display Jobs In Schedule)	2-357
	WT (Display Time)	2-358
	WX (Display Removed Space).	2-359
	WY (Display Job Status).	2-360
	XA (Add/Remove A Channel From An Exchange)	2-362
	XC (Inhibit/Uninhibit Or Display Status Of A Channel).	2-363
	XD (Remove Disk Segments).	2-364
	XM (Remove Memory From System)	2-365
	XP (Remove Pack Sectors).	2-366
A	USING RAILROAD SYNTAX DIAGRAMS IN THIS MANUAL	A-1
	Reading A Railroad Diagram.	A-1
	Items Of A Railroad Diagram	A-2
	Constants	A-2
	Variables.	A-2
	Required Items	A-3
	Optional Items	A-3
	Loops And Bridges	A-4
	Punctuation	A-6
	Conclusion	A-6
B	SYNTAX OF COMMANDS FOR USER-CODED LOADMP, PACKUP, PBDOUT, And PBTOUT UTILITIES	B-1
	GLOSSARY OF TERMS	Glossary-1
	INDEX	1

LIST OF ILLUSTRATIONS

Figure	Title	Page
Figure 2-1	AC Command Syntax (Format 1)	2-2
Figure 2-2	AC Command Syntax (Format 2)	2-2
Figure 2-3	AD Command Syntax	2-4
Figure 2-3-A	AD Specifications — AM, WM, MX Syntax	2-8
Figure 2-4	AFTER Command Syntax	2-10
Figure 2-5	AJ Command Syntax	2-12
Figure 2-6	ALLOCATE Command Syntax	2-13
Figure 2-7	ALTER Command Syntax	2-16
Figure 2-8	ALTER BACKUP Command Syntax	2-16-A
Figure 2-9	ALTER EXCHANGE Command Syntax	2-17
Figure 2-10	ALTER FILE Command Syntax	2-17
Figure 2-11	ALTER PACK Command Syntax	2-18
Figure 2-12	ALTER SYSTEM Command Syntax	2-19
Figure 2-13	AT Command Syntax	2-21
Figure 2-14	AT Command Syntax	2-22
Figure 2-15	AX Command Syntax	2-24
Figure 2-16	AX Command Syntax	2-25
Figure 2-17	BCL Command Syntax	2-27
Figure 2-18	BEGINUSER Command Syntax	2-28
Figure 2-19	BF,BFP Command Syntax	2-30
Figure 2-20	BFRF Command Syntax	2-32
Figure 2-21	Executing BFRF With The EXECUTE Command	2-37
Figure 2-22	BINARY Command Syntax	2-38
Figure 2-23	BO Command Syntax	2-39
Figure 2-24	BT Command Syntax	2-40
Figure 2-25	BUFFER Command Syntax	2-42
Figure 2-26	CANDIDATE Command Syntax	2-43
Figure 2-27	Response to CANDIDATE Command	2-43
Figure 2-28	CC Command Syntax	2-45
Figure 2-29	CD Command Syntax(Format 1 - Disk)	2-47
Figure 2-30	CDP Command Syntax(Format 2 - Pack)	2-47
Figure 2-31	CHANGE Command Syntax	2-49
Figure 2-32	CHARGE Command Syntax	2-52
Figure 2-33	CK Command Syntax	2-54
Figure 2-34	CN Command Syntax	2-55
Figure 2-35	COMPILE Command Syntax	2-57
Figure 2-36	Syntax For <parameter list>	2-58
Figure 2-37	COMPILE Command With Slash Parameters Syntax	2-61
Figure 2-38	COPY,COMPARE, COPYADD, MERGE and MOVE Commands Syntax	2-63
Figure 2-39	Command Options Syntax	2-65

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
Figure 2-40	Boolean Options	2-65
Figure 2-41	File Specification Syntax	2-68
Figure 2-42	Volume Specification Syntax	2-69
Figure 2-43	Attributes Syntax	2-70
Figure 2-44	CV,CVP Command Syntax(MCP Version)	2-74
Figure 2-45	CV,CVP Command Syntax(User-Coded Version)	2-74
Figure 2-46	DA Command Syntax(Format 1 - Disk)	2-76
Figure 2-47	DA Command Syntax(Format 2 - Pack)	2-76
Figure 2-48	DATA/DATAB Command Syntax	2-78
Figure 2-49	DBCA Command Syntax	2-80
Figure 2-50	DBCL Command Syntax	2-82
Figure 2-51	DBDR Command Syntax	2-83
Figure 2-52	DBIC Command Syntax	2-84
Figure 2-53	DBMM Command Syntax	2-85
Figure 2-54	DBSR Command Syntax	2-86
Figure 2-55	DBUA Command Syntax	2-87
Figure 2-56	DBUL Command Syntax	2-88
Figure 2-57	DC Command Syntax (NON-ICM Type)	2-89
Figure 2-58	DC Command Syntax (NON-ICM Type)	2-89
Figure 2-59	DC Command Syntax (ICM Type)	2-90
Figure 2-60	DD Command Syntax (ICM Type)	2-91
Figure 2-61	DEBUG Command Syntax (ICM Type)	2-92
Figure 2-62	Syntax For <parameter list>	2-93
Figure 2-63	DISK Command Syntax	2-96
Figure 2-64	DIR Command Syntax	2-99
Figure 2-65	DISPLAY (Program,MCP,or Intrinsic Code File	2-101
Figure 2-66	DISPLAY (Disk or Pack File)	2-102
Figure 2-67	DISPLAY (Absolute Disk Sector)	2-103
Figure 2-68	DL Command Syntax	2-105
Figure 2-69	DLP Command Syntax (Non-SSP)	2-106
Figure 2-70	DLP Command Syntax (SSP)	2-108
Figure 2-71	DM Command Syntax (User Task)	2-110
Figure 2-72	DM Command Syntax (MCP MIX)	2-110
Figure 2-73	DP Command Syntax	2-112
Figure 2-74	DQ Command Syntax	2-113
Figure 2-75	DR Command Syntax	2-114
Figure 2-76	DS Command Syntax	2-115
Figure 2-77	DX Command Syntax	2-118
Figure 2-78	EBCDIC Command Syntax	2-119
Figure 2-79	ED Command Syntax	2-120
Figure 2-80	END/ENDCTL	2-121
Figure 2-81	ENDAT	2-122
Figure 2-82	ENDUSER Command Syntax	2-123

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
Figure 2-83	ES Command Syntax	2-125
Figure 2-84	ET Command Syntax	2-126
Figure 2-85	EXECUTE Command Syntax	2-127
Figure 2-86	Syntax For <parameter list>	2-128
Figure 2-87	EXECUTE Command With Slash Parameters Syntax	2-130
Figure 2-88	FI Command Syntax	2-131
Figure 2-89	FILE Command Syntax	2-132
Figure 2-90	UNIT Record Syntax (Printer).	2-137
Figure 2-91	FILE Command Syntax (Auto Print Assignment)	2-138
Figure 2-92	FM Command Syntax	2-141
Figure 2-93	FN Command Syntax	2-142
Figure 2-94	FP Command Syntax	2-143
Figure 2-95	FR Command Syntax	2-144
Figure 2-96	GENERATE Command Syntax	2-145
Figure 2-97	GO Command Syntax	2-147
Figure 2-98	GT Command Syntax	2-148
Figure 2-99	Begin/End-Point Specification Syntax	2-148
Figure 2-100	HL Command Syntax	2-150
Figure 2-101	HN Command Syntax	2-152
Figure 2-102	ID Command Syntax	2-154
Figure 2-103	IL Command Syntax (Non-Disk Files)	2-155
Figure 2-104	IL Command Syntax (Disk, Pack Files)	2-156
Figure 2-105	IN Command Syntax	2-157
Figure 2-106	INSERT Command Syntax	2-159
Figure 2-107	KA Command Syntax	2-162
Figure 2-108	KP Command Syntax	2-165
Figure 2-109	KS Command Syntax	2-166
Figure 2-110	KX Command Syntax	2-168
Figure 2-111	LABEL1 Command Syntax	2-170
Figure 2-112	LC Command Syntax	2-174
Figure 2-113	LD,LDP Command Syntax	2-176
Figure 2-114	LH Command Syntax	2-177
Figure 2-115	LI Command Syntax	2-179
Figure 2-116	LN Command Syntax	2-180
Figure 2-117	LO Command Syntax	2-182
Figure 2-118	LOCK Command Syntax	2-184
Figure 2-119	LP Command Syntax	2-185
Figure 2-120	MEMDUMP Command Syntax	2-186
Figure 2-121	MEMORY Command Syntax	2-187
Figure 2-122	MR Command Syntax	2-191
Figure 2-123	MX Command Syntax	2-193
Figure 2-124	NET Command Syntax (Display Status).	2-194
Figure 2-125	Response to NET Command	2-194

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
Figure 2-126	NET Command Syntax (Change Network Mode).	2-195
Figure 2-127	NL Command Syntax.	2-197
Figure 2-128	NT Command Syntax.	2-198
Figure 2-129	NW Command Syntax.	2-199
Figure 2-130	OF Command Syntax.	2-200
Figure 2-131	OK Command Syntax.	2-201
Figure 2-132	OL Command Syntax.	2-203
Figure 2-133	OL Command Syntax (Hardware Type).	2-204
Figure 2-134	OL Command Syntax (Disk/Pack).	2-206
Figure 2-135	OL PCR Command Syntax.	2-208
Figure 2-135a	OL NCP Command Response.	2-208-A
Figure 2-136	OT Command Syntax.	2-209
Figure 2-137	OU Command Syntax.	2-210
Figure 2-138	PA Command Syntax.	2-212
Figure 2-139	PACK Command Syntax.	2-213
Figure 2-140	PATCH Command Syntax (Running MCP Code File).	2-215
Figure 2-141	PATCH Command Syntax (Program or Non-Running Code File).	2-215
Figure 2-242	PATCH Command Syntax (File).	2-216
Figure 2-143	PATCH Command Syntax (Disk Sector).	2-216
Figure 2-144	Executing PATCHK.	2-219
Figure 2-145	PC, PCP Command Syntax (MCP Version).	2-221
Figure 2-146	PC, PCP Command Syntax (User-Coded Version).	2-222
Figure 2-147	PD Command Syntax.	2-224
Figure 2-148	PERFORM Command Syntax.	2-226
Figure 2-149	PG Command Syntax (Format 1).	2-228
Figure 2-150	PG Command Syntax (Format 2).	2-228
Figure 2-151	PM Command Syntax.	2-230
Figure 2-152	DMPOUT Command Syntax.	2-231
Figure 2-153	<DMPOUT parameters> Syntax.	2-231
Figure 2-154	<memory dump> Syntax.	2-232
Figure 2-155	DMPANL Command Syntax.	2-232
Figure 2-156	<table selection> Syntax.	2-233
Figure 2-157	<table list> Syntax.	2-234
Figure 2-158	<raw memory selection> Syntax.	2-236
Figure 2-159	<task selection> Syntax.	2-236
Figure 2-160	<task parameters> Syntax.	2-237
Figure 2-161	<memory dump> Syntax.	2-237
Figure 2-162	PO Command Syntax.	2-239
Figure 2-163	PP Command Syntax.	2-240
Figure 2-164	PR, PRM, PRP, PRS Command Syntax.	2-241
Figure 2-165	PRINT Command Syntax.	2-244
Figure 2-166	SEARCH <STRING SPEC> SYNTAX.	2-246
Figure 2-167	AX Response Syntax (Option 1).	2-247

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
Figure 2-168	AX Response Syntax (Option 2)	2-248
Figure 2-169	AX Response Syntax (Option 3)	2-249
Figure 2-170	PRIORITY, PRM, PRP, PRS Command Syntax	2-252
Figure 2-171	PS Command Syntax	2-254
Figure 2-172	QD Command Syntax	2-255
Figure 2-173	QT Command Syntax	2-256
Figure 2-174	QWIKPOOL Command Syntax	2-258
Figure 2-175	QWKMEM Command Syntax	2-259
Figure 2-176	RA Command Syntax	2-260
Figure 2-177A	RD, RDP Command Syntax (Format 1 - Disk)	2-261
Figure 2-177B	RD, RDP Command Syntax (Format 2 - Pack)	2-261
Figure 2-178	REMOVE Command Syntax	2-263
Figure 2-179	RERUN Command Syntax	2-266
Figure 2-180	RF, RFP Command Syntax	2-267
Figure 2-181	RM Command Syntax	2-269
Figure 2-182	RN, RNP Command Syntax	2-271
Figure 2-183	RO Command Syntax	2-273
Figure 2-184	RP Command Syntax (Format 1)	2-274
Figure 2-185	RP Command Syntax (Format 2)	2-274
Figure 2-186	RQ Command Syntax	2-276
Figure 2-187	RS Command Syntax (Remove Specific Job)	2-277
Figure 2-188	RS Command Syntax (Remove All Jobs)	2-277
Figure 2-189	RUN Command Syntax	2-278
Figure 2-190	RW Command Syntax	2-279
Figure 2-191	RX Command Syntax (Memory)	2-280
Figure 2-192	RX Command Syntax (Disk)	2-280
Figure 2-193	RXS Command Syntax (Pack)	2-281
Figure 2-194	RY Command Syntax	2-282
Figure 2-195	SB Command Syntax (Specifying Mix Number)	2-284
Figure 2-196	SB Command Syntax (Specifying Database Name)	2-284
Figure 2-197	SD Command Syntax	2-286
Figure 2-198	SHOW Command Syntax	2-287
Figure 2-199	SHOW BACKUP Command Syntax	2-288
Figure 2-200	SHOW BLT Command Syntax	2-288
Figure 2-201	SHOW EXCHANGE Command Syntax	2-288
Figure 2-202	SHOW FILE Command Syntax	2-289
Figure 2-203	SHOW IRS Command Syntax	2-289
Figure 2-204	SHOW PROCESSORS Command Syntax	2-289
Figure 2-205	SHOW SSP Command Syntax	2-289
Figure 2-206	SHOW TASK Command Syntax	2-290
Figure 2-207	SHOW TASKS Command Syntax	2-290
Figure 2-207a	SHOW TRAK Command Syntax	2-290
Figure 2-208	SI Command Syntax	2-291
Figure 2-209	SK Command Syntax	2-292

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
Figure 2-210	SM Command Syntax	2-293
Figure 2-211	SN Command Syntax (Format 1)	2-294
Figure 2-212	SN Command Syntax (Format 2)	2-294
Figure 2-213	SO Command Syntax	2-296
Figure 2-214	SPO Command Syntax	2-301
Figure 2-215	SQ Command Syntax	2-302
Figure 2-216	SQP Command Syntax	2-303
Figure 2-217	SS Command Syntax	2-306
Figure 2-218	ST Command Syntax	2-307
Figure 2-219	START Command Syntax	2-308
Figure 2-220	STOP Command Syntax	2-309
Figure 2-221	SV Command Syntax	2-310
Figure 2-222	SW Command Syntax	2-311
Figure 2-223	TEST Command Syntax	2-313
Figure 2-224	TI Command Syntax	2-314
Figure 2-225	TI Display	2-314
Figure 2-226	TIME Command Syntax	2-315
Figure 2-227	TL Command Syntax	2-316
Figure 2-228	TM Command Syntax (Write Tape Mark Only)	2-318
Figure 2-229	TM Command Syntax	2-318
Figure 2-230	TO Command Syntax	2-320
Figure 2-231	TR Command Syntax	2-321
Figure 2-232	TRACE Command Syntax	2-322
Figure 2-233	Begin/End-Point Specification Syntax	2-322
Figure 2-234	UL Command Syntax	2-324
Figure 2-235	UNIT Command Syntax (CRD And PCH)	2-327
Figure 2-236	UNIT Command Syntax (DCP)	2-328
Figure 2-236a	UNIT Command Syntax (NCP)	2-328-A
Figure 2-237	UNIT Command Syntax (IPP)	2-329
Figure 2-238	UNIT Command Syntax (ISC)	2-330
Figure 2-239	UNIT Command Syntax (MPE, GCR, MTC, MT9)	2-331
Figure 2-240	UNIT Command Syntax (NST)	2-332
Figure 2-241	UNIT Command Syntax (ODT, OCS)	2-333
Figure 2-242	UNIT Command Syntax (PRN, TPR)	2-334
Figure 2-243	UNIT Command Syntax (RJE, TC5, VDD, TWX)	2-335
Figure 2-244	UNIT Command Syntax (S4A, S4B)	2-336
Figure 2-245	UP Command Syntax	2-338
Figure 2-246	UR Command Syntax	2-339
Figure 2-247	USER Command Syntax	2-341
Figure 2-248	VALUE Command Syntax	2-343
Figure 2-249	WB Command Syntax	2-345
Figure 2-250	WC Command Syntax	2-346
Figure 2-251	WD Command Syntax	2-347

LIST OF ILLUSTRATIONS (Continued)

Figure	Title	Page
Figure 2-252	WD Display	2-347
Figure 2-253	WJ Command Syntax.	2-349
Figure 2-254	WJ Display	2-350-B
Figure 2-255	WM Command syntax	2-351
Figure 2-256	WM Display	2-351
Figure 2-257	WO Command Syntax	2-353
Figure 2-258	WO Display	2-353
Figure 2-259	WQ Command Syntax	2-355
Figure 2-260	WS Command Syntax.	2-357
Figure 2-261	WT Command Syntax	2-358
Figure 2-262	WX Command Syntax	2-359
Figure 2-263	WY Command Syntax	2-360
Figure 2-264	XA Command Syntax.	2-362
Figure 2-265	XC Command Syntax.	2-363
Figure 2-266	XD Command Syntax.	2-364
Figure 2-267	XM Command Syntax	2-365
Figure 2-268	XP Command Syntax	2-366
Figure B-1	ADD Command Syntax (TAPE to DISK)	B-1
Figure B-2	ADD Command Syntax (PACK to DISK)	B-2
Figure B-3	ADD Command Syntax (PACK Tape to PACK)	B-2
Figure B-4	ADD Command Syntax (DISK to PACK)	B-2
Figure B-5	ADD Command Syntax (PACK to PACK)	B-3
Figure B-6	ADD Command Syntax (DISK Tape to PACK).	B-3
Figure B-7	ADD Command Syntax (Pack Tape to DISK)	B-3
Figure B-8	CHECK Command Syntax	B-4
Figure B-9	DUMP Command Syntax (Disk)	B-4
Figure B-10	DUMP Command Syntax (Disk to Pack)	B-4
Figure B-11	DUMP Command Syntax (Pack to Tape)	B-5
Figure B-12	DUMP Command Syntax (Pack to Disk)	B-5
Figure B-13	DUMP Command Syntax (Pack to Pack)	B-5
Figure B-14	LOAD Command Syntax (DISK).	B-6
Figure B-15	LOAD Command Syntax (Tape to Pack)	B-6
Figure B-16	LOAD Command Syntax (Pack to Disk)	B-6
Figure B-17	LOAD Command Syntax (Disk to Pack)	B-7
Figure B-18	LOAD Command Syntax (Pack to Pack)	B-7
Figure B-19	LOAD Command Syntax (Tape to Pack)	B-7
Figure B-20	LOAD Command Syntax (Tape to Disk)	B-7
Figure B-21	UNLOAD Command Syntax	B-8
Figure B-22	UNLOAD Command Syntax (Pack to Tape)	B-8
Figure B-23	UNLOAD Command Syntax (Disk to Pack)	B-8
Figure B-24	UNLOAD Command Syntax (Pack to Disk)	B-9
Figure B-25	UNLOAD Command Syntax (Pack to Pack)	B-9
Figure B-26	PB Command Syntax	B-9



LIST OF TABLES

Table	Title	Page
1-1	Examples of File-Identifiers	1-9
1-2	Reserved File-Identifiers	1-10
1-3	Commands And Their Access Level Classifications	1-13
2-1	AD Command Definition	2-4
2-2	Table Display Options	2-7
2-2	Table Display Options	2-7
2-3	DISK/LAK Pack Maximum Address Ranges	2-97
2-4	Hardware Mnemonics	2-107
2-5	Recommended Channel Priorities	2-108
2-8	Hardware Media and Mnemonics	2-134
2-9	Allowable File Equates Across Hardware Types	2-135
2-10	Host Load Files And Their Usage	2-178
2-11	Hardware Mnemonics	2-205
2-12	System Options Available For Setting With SO	2-296
2-13	Memory Locations of Programmatic Switches	2-312
2-14	Memory Locations of Programmatic Switches	2-343



ABOUT THIS MANUAL

The *V Series MCP/VS System Software Operations Guide Volume 2: System Commands* describes the commands that allow you to interface with the Master Control Program (MCP) for the V Series computer systems. The various system commands allow inquiry or change of system, program, and device status; initiation and modification of system utilities; and initiation of programs from the user program library.

SCOPE AND PURPOSE

The *V Series MCP/VS System Software Operations Guide Volume 2: System Commands* lists the system commands in alphabetical order and provides detailed information about the use and syntax for each command. This manual provides some basic information for the first-time or occasion user as well as reference material for the experienced user.

The system commands documented in this manual operate under MCP/VS 2.0 on a V Series processor.

PREREQUISITES

This manual provides information about system commands for data management personnel, systems analysts, and systems operators. Some familiarity with data processing and Unisys V series systems is required.

ORGANIZATION OF THIS MANUAL

This manual consists of two sections, two appendixes, and a glossary, as follows:

SECTION 1: INTRODUCTION TO THE SYSTEM COMMANDS

This section discusses the different ways system commands are entered into the system to communicate with the Master Control Program (MCP); the distinction between keyboard input messages and MCP control instructions; some details that pertain to the commands in general; and command restrictions.

SECTION 2: THE SYSTEM COMMANDS

This alphabetical listing of the system commands describes and explains the syntax of each command.

APPENDIX A: USING RAILROAD SYNTAX DIAGRAMS IN THIS MANUAL

For those who are unfamiliar with railroad syntax diagrams, this appendix provides information on how to follow these diagrams.

APPENDIX B: SYNTAX OF COMMANDS FOR USER-CODED LOADMP, PACKUP, PBDOUT, And PBTOUT UTILITIES

The ADD, CHECK, DUMP, LOAD and UNLOAD commands invoke the LOADMP and PACKUP utilities. The functions of these utilities have been replaced by the SYSTEM/COPY utility. The PB command invokes the PBTOUT or PBDOUT utilities, which have been replaced by PBDPRN. Unisys no longer provides the LOADMP, PACKUP, PBDOUT, and PBTOUT utilities, but user-written versions can be invoked by the applicable commands. This appendix contains the syntax of these commands.

GLOSSARY OF TERMS

The glossary provides definitions of terms and concepts used throughout the *V Series MCP/VS System Software Operation Guide Volumes 1 through 4*.

RELATED PRODUCT INFORMATION

V Series MCP/VS System Software Operations Guide Volume 1: Installation

This first volume of the MCP/VS System Software Operations Guide contains introductory material about the MCP/VS 2.0 operating system and details the procedures and options available for installing and initializing MCP/VS 2.0 on V Series systems.

V Series MCP/VS System Software Operations Guide Volume 3: System Utilities

The MCP/VS 2.0 utilities act as extensions of the Master Control Program (MCP) for such functions as converting files from one medium to another, initializing pack units, or creating pseudo card decks.

V Series MCP/VS System Software Operations Guide Volume 4: Output Messages

The MCP/VS System Software output messages are those messages that appear on the Operator Display Terminal (ODT) to either inform you of a process taking place or of an error condition.

V Series MCP/VS Architecture Manual

This manual explains and describes the structure of the the MCP/VS 2.0 operating system.

V Series BNA Installation and Operations Guide

This manual discusses the V Series Network Architecture (BNA) software, which links V series computer systems together and to other Unisys computer systems in a BNA data communications/distributed processing network configuration.

V Series CANDE Installation and Operations Guide

CANDE is an acronym for the Command And Edit timesharing system. CANDE provides interactive generalized file preparation and updating capabilities and task control from your terminal.

V Series COBOL ANSI-74 Compiler Programming Reference Manual

This manual discusses the COBOL-74 (Common Business Oriented Language ANSI-74 Standard) Language as implemented on Unisys V Series systems.

B 2000/B 3000/B 4000/V Series DMSII Operations Reference Manual Volume 1: DASDL Specifications

Volume 1 contains information on how to set up a database with the Data and Structure Definition Language (DASDL). The manual includes sections about creating a database, creating and compiling a DASDL source file, logical data structures within DMSII, DASDL syntax, and examples of DASDL specifications and DASDL database definitions.

B 2000/B 3000/B 4000/V Series DMSII Operations Reference Manual Volume 2: Administration

Volume 2 contains information the Database Administrator can use to customize or streamline the DMSII product for a particular environment. The manual includes sections about updating and reorganizing a database, audit and recovery functions, production data programs, an MCP interface to DMSII, database dumping and recovery, the database program, exception condition categories, and output messages.

B 2000/B 3000/B 4000/V Series DMSII Operations Reference Manual Volume 3: Utilities

Volume 3 contains all of the information for the following DMSII utilities: DMSUTL, DBPDMP, DMSTAT, DMSLST, DMSCTL, DMSDAP, DDFLIS, DMSAUD, DMSDLR. Each section covers one utility and the last section covers all utility messages.

V Series MCP/VS System Software Operations Guide, Volume 2 About This Manual

B 2000/B 3000/B 4000/V Series GEMCOS Implementation and Programming Reference Manual

This manual describes the features, methods of application, and utilities of the Generalized Message Control System (GEMCOS).

V Series System Software Logging Operations Reference Manual

This manual describes the components of the system ODT log (SLOG), Run log (RLOG), and Maintenance log (MLOG). These logs are produced by the Master Control Program (MCP) and can be analyzed by system or user-written utilities.

V Series Program Interfaces Programming Reference Manual

This manual describes the various programmatic interface areas available with MCP/VS 2.0

V Series MCP/VS Programming Reference Manual

This document provides advanced level information a programmer/analyst might need to know about MCP/VS 2.0, such as how to read a dump or write a user-coded utility program.

B 2000/B 3000/B 4000/V Series PROPS Operations Services Operations Guide

This manual describes how to use the Programmer's Productivity System (PROPS) services. These services benefit both programmers and operators in providing electronic mail capabilities, system log search functions, and the ability to initiate a job and monitor all ODT messages relating to the job on the terminal.

B 2000/B 3000/B 4000/V Series Remote Job Entry (RJE) Operations Guide

Remote Job Entry (RJE) is a software package that allows jobs, data, and control commands to be sent from a remote site to a central system, and that allows output of data from the central system to be sent to remote peripherals.

V Series System Software Security Installation and Operations Guide

The system security facilities for the Unisys V series systems allow access control to the system and to files.

B 2000/B 3000/B 4000/V Series WFL Compiler Programming Reference Manual

The Work Flow Language (WFL) is a Unisys language used for constructing jobs that compile and run programs on Unisys computer systems. WFL includes variables, expressions, and flow-of-control statements that offer a range of task control capabilities.



SECTION 1

INTRODUCTION TO THE SYSTEM COMMANDS

SYSTEM COMMAND OVERVIEW

Systems operators and programmers can direct the MCP to perform particular actions through the use of Master Control Program (MCP) system commands. These commands have many functions, including display or change of system, program, or device status and initiation and control of various system utilities. This manual lists the system commands in alphabetical order and discusses the use and syntax of each command.

System commands can be entered in the following ways:

- At an Operator Display Terminal (ODT) or Operator Control Station (OCS).
- Through the programmatic ZIP and ZIPSPO facilities in the syntax of the different programming languages. Most programs can contain ZIP statements that pass system commands to the MCP.
- Through programmatic SPOMESSAGE Branch Communicates (SPOM BCTs) that request information from the MCP.
- Through a card reader or pseudo card reader.
- Through the command capabilities provided with some programming tool packages, such as the Command and Edit system (CANDE) and the Programmer Productivity System (PROPS).
- Through the Work Flow Language (WFL).
- Through the Remote Job Entry (RJE) system that allows communication between a remote site and a central system.

Keyboard Input Messages vs. MCP Control Instructions

There are two types of MCP system commands: keyboard input messages and MCP control instructions. Each of the commands described in this manual specifies whether it is a keyboard input message or an MCP control instruction.

Keyboard input messages have a wide range of functions, such as inquiry and system maintenance. For the most part, keyboard input messages, are two- or four-character abbreviations, such as DD (Delete DLP) or TR (Time Reset).

The MCP control instructions, generally, affect a program in the mix, in that they either initiate or provide special parameters for a program's execution. These control instructions are mostly full-word commands, such as EXECUTE (Execute Library Program) or PERFORM (Schedule DMPALL).

The precise differences in how the two are entered are as follows:

- The SPOMESSAGE BCT passes only keyboard input messages to the MCP, not MCP control instructions.
- A keyboard input message entered from a card reader or in a programmatic ZIP must be prefaced by SPO.
- When you enter commands through a ZIP statement within a program, the syntax among the different programming languages does vary, but MCP control instructions are usually preceded by ZIP, keyboard input messages by ZIP SPO.
- A keyboard input message entered through CANDE or a programmatic SPOMESSAGE BCT displays on the ODT as "==> SPO" followed by the text that was sent. An MCP control instruction entered through CANDE or RJE, displays as "==>" followed by the text you sent.
- All MCP control instructions supplied through the keyboard can optionally be prefaced with the CC (Enter Control Instructions) command (for example, CC EXECUTE MYPROG).
- If you are using BNA network architecture, note the following:
 - If a system running MCP/VS 2.0 is communicating with a system running MCP/VS 1.0 through BNA, the CC is mandatory when using MCP control instructions after the AT (Initiate BNA Job Transfer) command, which initiates a BNA job transfer between a local host and a remote host.
 - The keyboard input messages that start a job must be prefaced by the word SPO after the AT command. This includes the CV (Convert Punch Backup File To Pseudo Card Deck), KA (Analyze Disk Directory), KP (Print Disk Segments), KS (Analyze Disk Space), PA (Analyze Pack Directory), PB (Print Backup File), PC (Create Punch Backup File), PM (Print Memory Dump), PP (Print Pack Sectors), and PS (Analyze Pack Space) commands.

ENTERING COMMANDS THROUGH THE ODT

System commands are entered at the top line of the ODT, and they can continue to subsequent lines.

To enter commands through the ODT, do the following:

1. Position the ODT cursor at the top of the screen by pressing the CLEAR/HOME key, if necessary.
2. Type the desired command.
3. If the ODT is a B 20, ET 1100, TD 830, or MT 983 device, you can transmit the command by pressing the transmit key. All data preceding the cursor will be transmitted to the MCP. For other devices, press the End-Of-Text (ETX) key, which places an ETX character on the screen and returns the cursor to the Home position. Then press the transmit key.

Optionally, the command can be terminated with a period (.) character prior to the ETX, in which case all text following the period is ignored. When the MCP acknowledges the command, the ODT is set in Receive mode.

If you have multi-line input, put an ETX character at the end of the message; put the cursor at the beginning of the message; then press the transmit key.

NOTE

Pressing the transmit key causes everything from the ODT cursor to the ETX character to be sent to the MCP. It is important, therefore, that the cursor be positioned at the beginning of the message before the transmit key is pressed and that no spurious ETX characters be on the screen.

If an error is made while typing in a command, press the CLEAR HOME key and then retype the command.

If a command is entered incorrectly, the MCP returns the message:

****KBD IGNORED: INV SYNTAX**

MCP Response To ODT Input

The MCP responds in the following way to ODT keyboard input:

A single-line response to an input request will be displayed on the second line of the screen (the remainder of the screen will not be cleared). For a multiple line (or no-line) response, the second line of the screen will be cleared. A multiple line response will be displayed following the header and any skip-lines.

System commands are indented one additional space when they are displayed on the ODT "messages" screen and in the ODT log listing to make it easier to identify input requests. However, this does not affect keyboard output messages returned to a SPOMESSAGE BCT.

The system message display is indented four spaces. The lines that have not previously appeared in a display will be highlighted, independently of system message displays on any other ODT terminal. It will therefore be possible to see at a glance which messages are new.

When input messages are entered from an ODT, they appear in a DQ (Display System Messages) command response and/or from the AD MSG display. All input messages are flagged with "==" to the left of the display line.

If the command requires the MCP to display multiple lines, all available lines except the communication interface area and lines reserved by the ODT SK option are used, and lines currently being displayed are overwritten or erased. See UNIT Record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*, and the AD (Specify ODT Screen Parameters) command for a description of the ODT SK, HDR, and MSG options.

Replies that exceed one line remain on the screen for 30 seconds or until the system operator enters an ETX (or transmits blanks) for a new screen display, an ES (End Screen – ODT Only) command to end screen display, or another input request that would cause the current display to be overwritten or erased.

If the reply to the command requires more lines than are available on the ODT screen, the MCP uses all available lines except the communication interface area lines and the lines reserved by the ODT SK option. The MCP then displays "MORE DATA" in the communication interface area. No more data is written on the screen for 30 seconds or until the system operator enters an ETX, blank(s), ES, or another keyboard input request. If, after 30 seconds, the system operator has not performed any of these operations, the MCP displays a new screen containing the overflow data.

ENTERING COMMANDS THROUGH A PROGRAMMATIC ZIP OR ZIPSPO

The commands sent with the ZIP or ZIPSPO facilities are acted on exactly as if they had been entered at the ODT. The ZIP and ZIPSPO facilities are invoked by constructs in the various programming languages, which is discussed in the various language manuals.

If a program under execution contains a ZIP or ZIPSPO statement or equivalent syntax in a programming language that performs communication with the MCP, the statement must refer to a data area within the program. The information in that data area must conform to the following rules:

- Unless the first character of the command is a period, the data area is assumed to be 73 characters long. The text can be up to 72 characters long, followed by a period. If a period is not present, the MCP will insert one in the 73rd character position.
- If the first character of the command is a period, the data area can be of any length, and the MCP will not insert a period. The end period must therefore be present in the program.
- The hyphen (not enclosed in parentheses) cannot be used as a continuation character in a ZIP statement.
- The program must be able to handle four-digit mix numbers.

Note that MCP/VS commands not supported by MCP/VS cannot be used in a ZIP statement.

Entering Commands Through A ZIPSPO

The ZIPSPO function gives a user program the ability to pass control information to the MCP and to determine if any errors occurred during the processing. If there is an error, a response will be returned to the program.

The ZIPSPO response area is 82 characters long. The first two characters contain a numeric error code; the remainder contains the text.

If no errors are detected, the error code will contain "00" followed by a four-character value containing the Run log number of the last job initiated by the ZIPSPO or "0000" if no jobs were initiated. The remaining 76 characters are undefined. If an error is detected, the first two characters are non-zero and the following 80 characters will be a "***ZIP IGNORED: ..." keyboard response terminated by ETX.

ENTERING COMMANDS THROUGH A SPOMESSAGE BCT

The SPOMESSAGE BCT passes keyboard input messages to the MCP and requests that the response be returned to the requestor. The input message must be terminated by either a period (.) or an ETX character.

Note that you cannot enter MCP control instructions through a SPOMESSAGE BCT.

The length of the response buffer must be at least 80 characters and must not overlap the input area. The output buffer should be cleared before calling this BCT.

If you enter a command not valid through a SPOMESSAGE BCT, a message stating that the request is not allowed appears on the ODT.

See the *V Series Program Interfaces Programming Reference Manual* for more information.

ENTERING COMMANDS THROUGH PUNCHED CARDS

If punched cards are used to communicate a command, the following rules apply:

- The first column must contain an invalid character, known as the Punched Cards Control Character. This control character is a punch combination that does not represent any valid character. For example, multi-punched 1-2-3 is most commonly used. Invalid characters are not allowed in any other column. The first word following the invalid character must be a valid command. When a command is contained in a pseudo-card file, the first column must contain a question mark character ("?", EBCDIC 6F).

Only columns 2 through 72 can contain commands; the MCP ignores information after column 72. Commands can start in any position between columns 2 through 72.

- If the special character period (.) appears in a control record, other than as part of a file name in parentheses, such as (B974.1), all information following the period is ignored for control purposes. This allows comments to be present in control records.
- The special character semicolon (;) must be used to designate the end of a series of commands if a new command is to start on the same punched card or if the MCP would not otherwise recognize the new control data. This is used primarily to enter control data such as CHARGE, PRIORITY, MEMORY, and so forth, to a system intrinsic such as DMPALL. The semicolon can be used on a ?SPO record to designate multiple commands (for example, ?SPO KA =; MEM + 20 PR 5).
- Commands can be contained on more than one record, except for the ?SPO commands, which must always be contained on a single record. The continuation character hyphen (-) can be used to continue information from one record to another. The hyphen must not be used to divide any word. Any information following the hyphen (on the same record) is ignored. The record on which the information is continued must not contain an invalid character in column 1.

- All commands are described on the following pages under headings that appear to indicate that each of them must consist of a separate record (a COMPILE record, a PRIORITY record, and so forth). However, if the text of a command is delimited by a semicolon, indicating the logical end of that command, it can then be followed by another command on the same record.
- Any program name or file name that begins with a number or contains special characters must be enclosed in parentheses.
- The MCP treats every program name and file-identifier as if it had six characters, and shorter identifiers are assumed to have an appropriate number of trailing blanks. Consequently, any program name or file-identifier that contains special characters or embedded blanks must be enclosed in parentheses. It is syntactically correct to enclose only the actual identifier within parentheses. For example, (X-X) and (X-X) are both correct.

ENTERING COMMANDS THROUGH CANDE, PROPS, OR WFL

The Command And Edit (CANDE) system is a multi-user timesharing facility that provides several levels of system and file access control. If you have CANDE installed at your site and want to enter commands through CANDE, see the *V Series CANDE Installation and Operations Guide* for detailed information.

The Programmer's Productivity System (PROPS) is an interactive program development environment. If you have PROPS installed at your site and want to enter commands through PROPS, see the *B 2000/B 3000/B 4000/V Series PROPS Operations Services Operations Guide* for detailed information.

The Work Flow Language (WFL) is a Unisys language used for constructing jobs that compile and run programs on Unisys computer systems. WFL includes variables, expressions, and flow-of-control statements that offer a range of task control capabilities. See the *B 2000/B 3000/B 4000/V Series WFL Compiler Programming Reference Manual* for detailed information about WFL.

ENTERING COMMANDS THROUGH RJE

The Remote Job Entry (RJE) system is a software package that permits commands to be sent from a remote site to a central system, and that allows output of data from the central system to be sent to remote peripherals.

See the discussion of RJE in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

For more information on the operation of RJE, see the *B 2000/B 3000/B 4000/V Series Remote Job Entry (RJE) Operations Guide*.

COMMAND DETAILS

The following comments apply generally to all the system commands:

- If the security option is set, any input device (ODT, card, reader, CANDE station) must be logged in with a valid access code (which can include a user code, password, and charge number) before commands can be entered through that device. Pseudo-card files must contain LI, USER, or BEGINUSER commands. See the *V Series System Software Security Installation and Operations Guide* for more information.
- Leading zeros are generally optional unless otherwise specified.
- The words and numbers in a system command are separated by delimiting characters, which are either blank spaces, commas, or semicolons. A space is sufficient in most cases.
- Any number of spaces may be placed between words without affecting the meaning. For example, the following are equivalent:

COPY A,B TO TAPE

COPY A , B TO TAPE

KS =; MEM +10

KS = ; MEM + 10

- Parentheses must be used to enclose identifiers containing non-alphanumeric characters.
- Up to 400 bytes of input text can be entered from an ODT. It is not possible for a word to wrap around from one line to the next because the terminal automatically inserts a CR/LF at the end of each line transmitted.
- If a system running MCP/VS 2.0 is communicating with a system running MCP/VS 1.0, system commands must be prefaced with the characters CC if they are entered through BNA through an ODT mechanism and start a job (that is, ADD, CHECK, COMPILE, COPY, CV, DIR, DUMP, EXECUTE, GENERATE, KA, KP, KS, LOAD, MERGE, MOVE, PB, PC, PERFORM, PM, PP, PRINT, PS, and UNLOAD).
- You cannot use a mix number of 0 or 1 in a system command, except for the DM (Dump And Continue) command, which allows 0 DM. Mix numbers 0 and 1 are reserved for the use of MCP/VS 2.0. If you do enter a mix number of 0 or 1, an error message will be displayed.

File-Identifiers

A file-identifier is the name contained in the directory for the referenced device, such as a tape, disk, or pack. A file-identifier can be up to six characters long, but cannot be made up of all nulls, all blanks, or all zeros.

All library maintenance commands use an expanded method of identifying a file-identifier. A file-identifier can contain any combination of EBCDIC characters with the exception of left and right parentheses, period, comma, semicolon, hyphen, slash, and space; however, these characters (except the right parenthesis) can be used if they are enclosed in parentheses.

An equal sign not enclosed in parentheses, used in any position of a file-identifier, indicates that any character is acceptable in that character position. An equal sign used in this fashion is also called a masking character.

A file-identifier that contains masking characters is referred to as a file-group-identifier. Trailing equal signs are assumed if the file-identifier ends with an equal sign.

The following examples, as shown in Table 1-1, are specific file-identifiers that would be syntactically equal to the file-group-identifier.

Table 1-1. Examples of File-Identifiers

File-Group Identifier	Matching File-Identifiers	Remarks
A=	A ABC ASMBLR	All file-identifiers starting with the letter A.
A		A Only.
=A=	AA BA ZAYZ	All file-identifiers having an A in the second character position.
A====F	ABCDEF AWXYZF	All file-identifiers with an A in the first character position and an F in the sixth.

An equal sign must not be separated by a blank space from the letters of a file-group-identifier. A single equal sign normally causes all file identifiers to be specified. Exceptions are detailed under each applicable system command.

The file-identifiers listed in Table 1-2 are reserved for use by the MCP and cannot be used for user-created disk files.

Table 1-2. Reserved File-Identifiers

ADFL*W	AD rule overflow file (* is a system number and can be 0 through 3)
CONFIG*	System Configuration File (* is a system number and can be 0 through 3)
CTLDCK	LDCNTL card deck
DSKAVL	Disk available table
DSKDIR	Disk directory
hlparm	Halt/Load Parameters
MCPMCP	Operating MCP area
mstavl	Master Available Tables
nLOG*w	MCP Logs (system and option dependent) (n denotes the name of the log and can be M, R, S) (* is a system number and can be 0 through 3) (w denotes a rotating log file number and can be 0 through 9, X. 0 is the current log file and X is a reserved log file.)
PCRXRF	Pseudo reader dictionary
pntp*t	Program Name Table (* is a system number and can be 0 through 3)
SQARCY	SQUASH Recovery file
USERFL	System Access Control file
USRCOM	System Access Control file
\$*0001	MCP memory dump area (* is a system number and can be 0 through 3)

Special Characters

The special characters listed below cannot be used in a file-identifier except when enclosed in parentheses. When not enclosed in parentheses, these special characters have the following interpretation:

Period (.)

delimits an ODT message line or a system command on one record.

Semicolon (;)

denotes another system command to immediately follow.

Comma (,)

delimits a file-identifier.

Hyphen (-)

indicates command continuation.

Equal (=)

indicates a masking character position.

Be careful when using file-identifiers that begin with special characters that are also used by the MCP, such as the following:

@, which identifies a printer backup file

#, which identifies a pseudo card deck

*, which identifies a punch backup file

\$, which identifies a dump file

NOTE

Files named ":" followed by five numbers (for example, ":03F01") will be removed from disk after every halt/load.

COMMAND RESTRICTIONS

The following restrictions apply to the various system commands:

Invalid from a programmatic ZIP or ZIPSPO:

ALLOCATE, ALTER, BCL, BINARY, BO, BT, DATA/DATAB, DD, DEBUG, DISK, DISPLAY/WHATS, DL, DLP, DQ, EBCDIC, ED, ES, ET, HL, ID, KX, LABEL1, LH, LO, PACK, PATCH, QD, QWIKPOOL, QWKMEM, SHOW, UNIT, XA.

The REMOVE === (remove all files) command cannot be ZIPed since it requires a null mix.

Invalid from program for a SPOMESSAGE BCT:

AC, AD, AT, BO, BT, CV, DD, DL, DM, DP, DR, DS, ED, ES, ET, FM, FP, FR, HL, ID, IL, KA, KP, KS, KX, LD, LH, LN, LO, MR, NET, NW, OF, OK, OU, PA, PB, PC, PG, PM, PO, PP, PR, PS, QD, QT, RA, RM, RN, RP, RQ, RS, RW, RX, RY, SD, SI, SM, SN, SP, ST, SV, TL, TM, TR, UL, XA, XD, XM, XP

Invalid through Card Reader (SPO commands):

BO, BT, DEBUG, DQ, ED, ES, ET, ID, KX, LI, LO, QD, XA.

Invalid from WFL handler:

ADD, CHECK, DUMP, LOAD, PRIORITY, PR, PRM, PRP, PRS, UNLOAD

Invalid through Remote Job Entry (RJE):

AC, AD, ADD, ALLOCATE, ALTER, AT, BT, CHECK, CK, DB, DD, DISK, DISPLAY/WHATS, DL, DLP, DR, DUMP, DX, ED, ES, ET, HL, ID, IR, KX, LABEL1, LD, LH, LN, LOAD, NET, NW, PACK, PATCH, PG, PO, PR, PRIORITY, PRM, PRP, PRS, QD, QWIKPOOL, QWKMEM, RO, RP, RQ, RW, RX, RY, SD, SHOW, SI, SN, SO, SP, SQ, SV, TL, TM, TR, UNIT, UNLOAD, UR, XA, XC, XD, XM, XP

Valid for MCP mix number 0:

DM

Access Level

A terminal can be restricted to allow input of only certain types of system commands. These restrictions can be set in the LEVEL parameter of the UNIT ODT record of the system configuration file, which declares the terminals that communicate with the MCP. See *V Series MCP/VS System Software Operations Guide Volume 1: Installation* for more information.

Table 1-3 shows the type and access level of each command.

Note that all commands permitted to lower-level ODTs are also permitted to ODTs with higher classifications. Except for level 8 or 9 ODTs, commands requiring a <mix no> can be used only if the <mix no> refers to a program originating from the station that also originated the command.

V Series MCP/VS System Software Operations Guide, Volume 2
Introduction to the System Commands

Table 1-3. Commands And Their Access Level Classifications

Type	Inquiry	Job Init./ Term.	Job Maint.	Disk Maint.	Periph Maint.	System Maint
Level	0	1	3	5	7	8
Cmds Allowed	AJ	AFTER	FILE	ADD	AC	ALLOCATE
	AT	AX	FN	CHANGE	AD	ALTER
	AX	CD,CDP	GO	CHECK	CK	BT
	BCL	CHARGE	GT	COMPARE	CN	DBCA
	BEGINUSER	COMPILE	IN	COPY	FM	DBCL
	BF	DC	MEMORY	COPYADD	LD	DBDR
	BFRF	DEBUG	NL	CV	PG	DBIC
	BINARY	DM	NT	DA	PO	DBMM
	BO	DP	OT	DIR	RP	DBSR
	BUFFER	DS	PR	DISPLAY/	RW	DBUA
	CA	EXECUTE	PRIORITY	WHATS	RY	DBUL
	CC	FP	PRM	DUMP	SN	DD
	DATA/B	FR	PRP	IL	SV	DISK
	DQ	GENERATE	PRS	KA	TM	DL
	EBCDIC	INSERT	SHOW	KP	UL	DLP
	END/	LC	SK	KS		DR
	ENDCTL	LOCK	ST	LOAD		DX
	ENDAT	LP	START	MERGE		ED
	ENDUSER	MEMDUMP	STOP	MOVE		ET
	ES	OF	TRACE	MR		HL
	FI	OK		OU		ID
	HN	PB		PA		IR
	KX	PC		PATCH		LH
	LABEL1	PERFORM		PP		LN
	LI	PM		PS		NET
	LO	PRINT		RD		NW
	MX	QT		REMOVE		PACK
	OL	RA		RF		QD
	PD	RERUN		RM		QWIKPOOL
	SM	RN		UNLOAD		QWKMEM
	SPO	RS				RO
	SS	RUN				RQ
	TI	SW				RX
	TO	TEST				SB
	USER	TIME				SD
	WB	UP				SI
	WC	VALUE				SO
	WD					SP
	WJ					SQ

V Series MCP/VS System Software Operations Guide, Volume 2
Introduction to the System Commands

Table 1-3. Commands And Their Access Level Classifications (continued)

Type	Inquiry	Job Init./ Term.	Job Maint.	Disk Maint.	Periph Maint.	System Maint
Level	0	1	3	5	7	8
Cmds Allowed	WM WO WQ WS WT WY					SQP TL TR UNIT UR WX XA XC XD XM XP

SECTION 2

THE SYSTEM COMMANDS

This section contains the system commands listed in alphabetical order. The description for the commands, in most cases, includes the following:

- Whether the command is a keyboard input message or an MCP control instruction
- The ODT access level of the command
- Introductory material stating the function of the command
- A railroad syntax diagram of the command. If you do not know how to interpret railroad syntax diagrams, see Appendix A.
- An explanation of the element(s) in the diagram, followed by command details or restrictions
- An example of the command
- References to other relevant commands and documents

AC (PURGE AND PLACE TAPE NUMBER IN LABEL)

Keyboard Input Message

Access Level: 7

The AC command purges a tape and puts a permanent identifying external physical tape number into a magnetic tape label.

The syntax for the AC command is shown in Figure 2-1. However, the AC command also has additional capabilities that allow you to assign the tape to a mix number when purged and/or to override a program's tape type request. The syntax for these capabilities is shown in Figure 2-2.

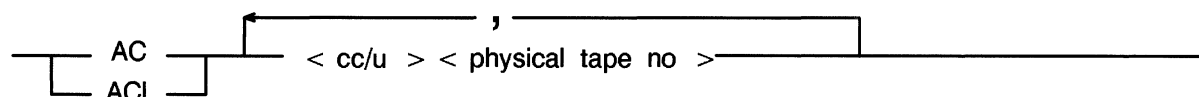


Figure 2-1. AC Command Syntax (Format 1)

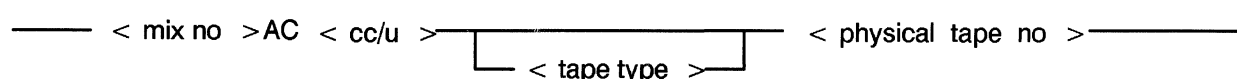


Figure 2-2. AC Command Syntax (Format 2)

Format 1

ACL

locks the tape after purging it.

<cc/u>

is the channel number/unit number of the appropriate device. The tape must be ready and contain a write ring.

<physical tape no>

is placed into the label of the designated tape reel and can be any integer up to five digits in length.

The <physical tape no> remains in the label throughout the life of the tape or until you perform another AC or SN (Purge And Place Tape Number In Label) operation on that tape.

Format 2

<mix no>

causes the tape on the drive designated by <cc/u> to be purged and assigned to the job with this mix number.

If the tape has GCR capability and no tape type is specified, GCR mode will be used.

<tape type>

overrides a program's tape type request (allowed only when <mix no> AC is used). The <tape type> can be MT9, MPE, GCR or MTC.

Details

If you use Format 1 (that is, omit the <mix no> and <tape type>) *and* you are using a GCR-capable drive, the tape will be purged and made available in GCR mode.

Examples

ACL 4/3 4

purges the magnetic tape on channel number 3/unit number 4, places the number 4 in its label, and then locks the tape.

AC 4/3 12345, 4/2 54321

purges the magnetic tape on channel number 4/unit number 3 and assigns it the number 12345; purges the magnetic tape on channel 4/unit number 2, and assigns it the number 54321.

14 AC 6/3 GCR 84127

purges the GCR magnetic tape on channel 6/unit number 3, assigns it to mix number 014, and places the number 84127 in its label.

See Also:

CN (Display Tape Number); LABEL1 (Change Tape Label); SN (Purge And Place Tape Number In Label); and TM (Write Tape Mark) commands.

AD (SPECIFY ODT/OCS SCREEN PARAMETERS)

Keyboard Input Message

Access Level: 7

The AD command controls the automatic display of system status information on an ODT or OCS by adding or deleting parameters.

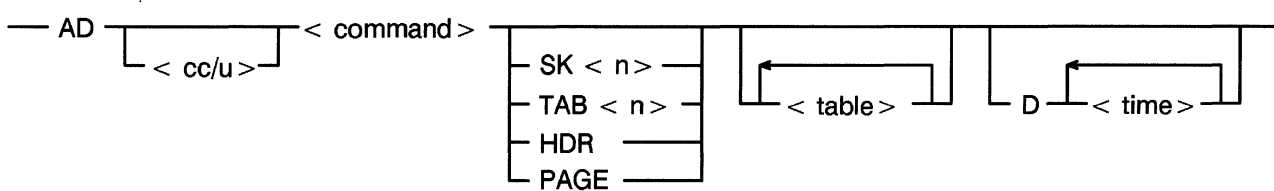


Figure 2-3. AD Command Definition

<cc/u>

is used to designate parameters on other ODT or OCS devices.

<command>

indicates one of the keywords defined in Table 2-1.

Table 2-1. AD Command Definition

Command	Meaning
ST	Stops the automatic display.
GO	Resumes the automatic display.
ADD	Adds the following table display data to the current table display rules. Replaces any duplicate table entries with the new rules.
DEL	Deletes the following table display rules. Only the table name is required. Multiple table names can be deleted. A group of table names enclosed in parentheses can be deleted provided the number of lines, n, and time, if specified, for the group are the same as those items to be deleted.
RPL	Replaces the current display rules by the following table display data.

Table 2-1. AD Command Definition (continued)

Command	Meaning
DEF	Applies default automatic display rules for the device. The default displays three sets of tables. The first set of tables lists information about the jobs active in the mix, jobs waiting, and the jobs in the schedule. These tables appear on the left side of the first screen of information. The second set of tables has four tables that list information regarding the status of the disk, pack, tape, and printer devices on the system. These tables appear on the right side of the first screen of information. Finally, the third set of tables lists the latest commands that have been input to the MCP and any system output messages. These tables are on the second screen of information. The displays are formatted so that the first and second line are blank. The third line displays a heading with the system name, available memory and CPU, date, and time. The tables are displays below that line for 10-second durations. Finally, the PAGE option is set. This allows any excess information that could not fit on the screen to be displayed when the space bar and transmit key are pressed. The syntax that DEF represents is as follows: SK 1 HDR PAGE (AM WM S) (DSK DPK MTP PRN) D 10MSG D 10
DIS or blank	Displays the current automatic display rules for the device.

NOTE

If the display has been previously stopped by an AD ST message, subsequent displays of the rules are appended by ...STOPPED.

The options following <command> specify table display data and are the same as the System Status Tables options described in *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

SK <n>

sets the number of lines down from the header or from the top of the screen if there is no header text will be displayed. <n> can be any number from 1 to 9. If you do not specify SK, the display starts on line 3 or after the header line.

TAB <n>

denotes the left margin of the table display. <n> represents a column number and must be a multiple of 8. The maximum value for <n> is 40. The default value is 24

HDR

displays the following information on line 3: host name, available memory, available CPU, the date, and time.

PAGE

allows a display to exceed one screen if there are more lines to display than will fit on one screen.

If PAGE is not specified, all table displays (except responses to operator inquiries) will be truncated to one screen.

If you specify PAGE, you can display the extra information by pressing the space bar and then the transmit or SPCFY key. The message "More Data" indicates that more information is available.

When two tables are displayed simultaneously, the table on the left can overflow to the right. This can happen whether or not this option is set. Once all the extra information has been displayed, normal table display resumes automatically.

D <time>

specifies the time interval in seconds between displays, where <time> can be any value from 1 through 99. The default for <time> is 10 seconds. Time intervals less than 3 to 5 seconds are not recommended due to excessive system overhead. If entries on a page have conflicting delay values, the value of the last entry is used.

<table>

determines which table(s) are to be displayed with action(s) as indicated. If multiple tables are to be displayed together, they must be enclosed in parentheses. The table display options are defined in Table 2-2. The parameter <n> indicates the number of lines of information to display. Including values is optional.

Table 2-2. Table Display Options

Option	Meaning
MSG <n>	displays the most recent commands input to the MCP along with any message the MCP may have issued. No more than 22 commands or responses can be displayed at one time. If you want to display less than 22 lines of text, indicate the number of lines you want by entering the number following MSG. Because this table is 80 characters wide, it cannot be displayed simultaneously with any other table. ■
S <n>	causes display of information about scheduled jobs. This information includes the processor and memory priorities, and memory in use, amount of time in the schedule, and after-mix number for each job.
DPK <n>	causes display of the status for all or <n> packs on the system.
DSK <n>	causes display of the status for all or <n> disks on the system.
MTP <n>	causes display of the status for all or <n> magnetic tape drives on the system.
PRN <n>	causes display of the status for all or <n> printers on the system.

AM, WM, MX Tables

Special syntax is used for the table selections AM, WM, and MX. These tables display information about jobs in the mix. The syntax for these table selections is shown in Figure 2-3a.

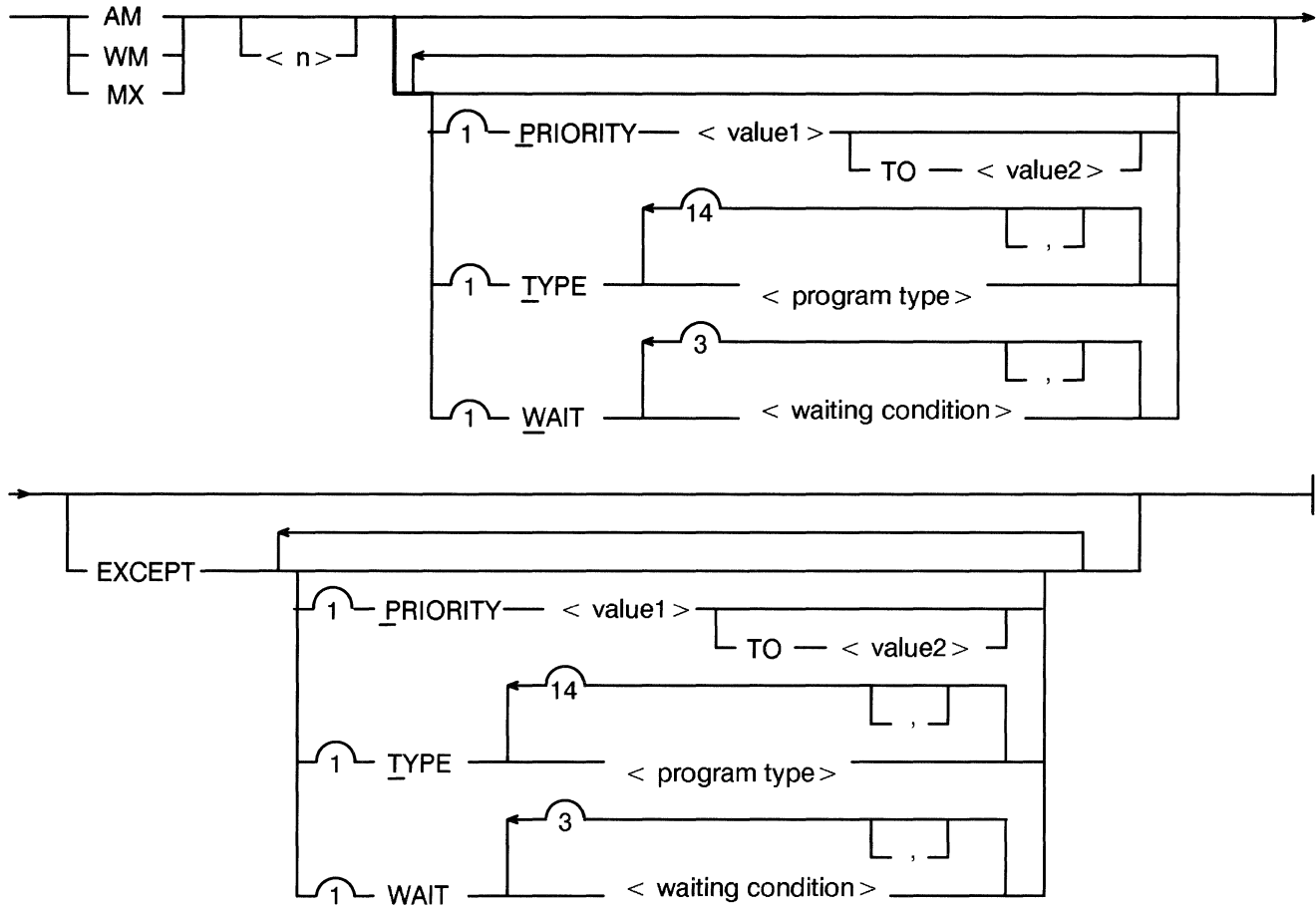


Figure 2-3a. AD Specifications — AM, WM, MX Syntax

AM
WM
MX

These three commands display information about jobs in the mix.

AM displays jobs in the active mix, including the task number, the processor priority, the memory priority, and the amount of memory in use for each job.

WM displays jobs in the mix that are waiting for a condition to be satisfied before processing resumes. The information displayed for each waiting job includes the processor and memory priorities and memory in use.

MX displays the status of the active mix and waiting mix at the same time.

<n>

Limits the lines of information, and consequently, the number of jobs displayed on the screen. <n> can range from 1 to 22; the default is 19.

PRIORITY <value1> T0 <value2>

Selects jobs based on processor priority. You have the option of selecting a single priority or a range of priorities. <value1> and the optional <value2> must be integers between 1 to 9. <value1> must be less than or equal to <value2>. The abbreviation P is accepted for PRIORITY.

TYPE <program type>

Selects jobs based on the type of program. The abbreviation T is accepted for TYPE. You can enter a maximum of 14 type selections. The available program types are:

<u>EXECUTE</u>	Program is executing
<u>COMPILER</u>	Program generates object code
<u>DMPALL</u>	Program is DMPALL intrinsic
<u>LOADMP</u>	Program is LOADMP intrinsic
<u>PACKUP</u>	Program is PACKUP intrinsic
<u>DSKOUT</u>	Program is DSKOUT intrinsic or diskpack SQUASH intrinsic
<u>MCS</u>	Program is Message Control System
<u>TIMESHARE</u>	Program is executed as time sharing task
<u>SHAREHNDL</u>	Program is a time sharing handler program (for example, CANDE)
<u>HANDLERS</u>	Program is any type of handler (BNA, WFL, time sharing)
<u>COMPSHAR</u>	Program that generates object code, executed as a timesharing task
<u>DMSPROG</u>	Program is a DMSII control program (for example, DBP)
<u>WFLHNDL</u>	Program is a Work Flow Language (WFL) handler
<u>BNAHNDL</u>	Program is a BNA network architecture handler
<u>COPY</u>	Program is SYSTEM/COPY intrinsic
<u>SORT</u>	Program is SORT: or SORT. intrinsic
<u>UTILITY</u>	Program is any type of intrinsic (DMPALL, LOADMP, PACKUP, DSKOUT, SYSTEM/COPY)

WAIT <waiting condition>

Selects jobs that are waiting for a specific condition to be satisfied (for example, waiting hardware, waiting memory and so forth). You can enter as many as 3 waiting conditions and 3 waiting condition exceptions. The abbreviation W is accepted for WAIT.

The available types of waiting conditions are:

<u>SPACE</u>	Waiting for memory, disk space or diskpack space
<u>DEVICE</u>	Waiting for hardware device to become available
<u>OPERATOR</u>	Waiting for operator to perform an action
<u>EVENT</u>	Waiting for a software event to be signalled (for example, STOQUE, CRCR and so forth)

EXCEPT

All text that follows the keyword EXCEPT is treated as exceptions to the selections entered previously. All of the exceptions must be grouped together after the keyword EXCEPT; you cannot enter a selection and an exception followed by another selection.

Details

All but one of the system status tables are 40 columns wide. Because of their size, the screen can be split to display two or more of these tables simultaneously. One or more tables can appear on one side of the screen, while one or more tables appear on the other side of the screen. Since the tables displayed by the MSG option are 80 columns wide, they require the entire screen.

When setting up the table displays, it is important to consider the amount of information the system has to display. For an open, readable display, group together the tables that have a limited amount of information. You could also limit the amount of information in that table.

To display more than one table at a time on the same half of the screen, enclose the options for those tables in parentheses. If the options are not in parentheses, the status tables are displayed one at a time.

Example

AD 6/3 RPL HDR (AM WM) (DSK DPK MTP)

replaces the current automatic display of the system status information for the OCS on channel 6, unit 3 with the following rules. There will be a header on line three. The active and waiting mix tables will be on one half of the screen, while the disk, pack, and tape status tables will appear on the other half.

See Also:

AJ (Display Active Jobs); ES (End Screen Display); MX (Display Jobs In Mix); OL (Display Peripheral Status); WJ (Display Waiting Jobs); WS (Display Jobs In Schedule); and WY (Display Job Status) commands.

The *V Series MCP/VS System Software Operations Guide Volume 1: Installation* for more information about system status tables.

ADD (Copy Files From Library Tape To Disk or Pack)

MCP Control Instruction

Access Level: 5

The ADD command copies files from a LOADMP or PACKUP library tape to disk or pack but does not replace an existing file.

It is recommended that you use the COPYADD option of the COPY command instead of the ADD command. The ADD command will become unavailable at a future release. See COPY (Invoke File Transfer Utility) for more information.

The ADD command starts a program called LOADMP, if the files are to be copied to disk, or a program called PACKUP, if the files are to be copied to pack. LOADMP and PACKUP must be written by the user; Unisys does not supply them. If the appropriate program is not present on disk, the ADD command is ignored.

See Appendix B for ADD syntax if you have user-coded LOADMP and/or PACKUP utilities.

See also:

COPY (Invoke File Transfer Utility) command. Appendix B.

AFTER (Hold Program)

MCP Control Instruction

Access Level: 1

The AFTER command holds execution of a program until the completion of a specified program.

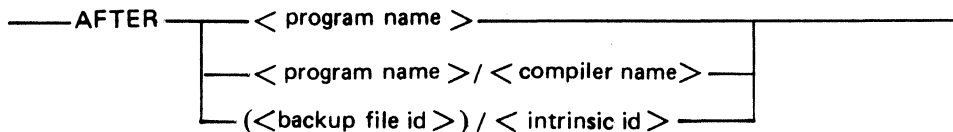


Figure 2-4. AFTER Command Syntax

<program name>

designates the name of an object code file. A program name can have up to six characters. The first character must be a letter of the alphabet.

<program name>/<compiler name>

can be used when a program is to be initiated AFTER a compilation and it is necessary to specify a particular compilation.

(<backup file id>)/<intrinsic id>

specifies an intrinsic. A printer backup file number is represented by an @ (at sign) followed by five digits.

If a program is to be initiated AFTER a program that has used the PROGRAM CALL BCT to call SORT; or MERG;, the AFTER command should specify only the program name, not <backup file id>/<intrinsic id>.

Details

When the MCP encounters AFTER in the control string, the MCP locates the highest process priority job in the mix with the requested name, <program name>, and attaches the job to be initiated to that program. If the specified job is not in the mix, the schedule is examined for the requested program (by schedule priority). If a match is not found, the first job entering the schedule with the requested name is used as the program, <program name>.

The program to be initiated is scheduled but not brought into the mix until <program name> has come to a normal End-Of-Job (EOJ).

There is no limitation on the length of time the program waits.

If <program name> or <program name>/<compiler name> does not go to a normal EOJ (if, for example, it has been discontinued by a DS (Discontinue Program) command or the compilation of <program name> contains syntax errors), the MCP does one of the following:

- If the AFTR option is reset (off), the program is left in the schedule until a program of the specified name reaches a normal EOJ. The operator can remove the job from the schedule with the RS (Remove Job From Schedule) command or allow the job to enter the mix with the RA (Remove After-Linkage) command.
- If the AFTR option is set (on), the program is automatically removed from the schedule. In addition, any jobs linked to this job through the AFTER command are removed from the schedule.

Examples

COMPILE PROGX COBOL9; AFTER PROGY

compiles PROGX with the COBOL9 compiler after PROGY finishes executing.

PFM DSKMTP SOLT SOURCE/TAPE 80 9; AFTER PROGX/COBOL9

initiates the DMPALL intrinsic to reproduce the data file called SOLT on disk to a tape called SOURCE and file named TAPE, with a record length of 80 and a block size of 9, after PROGX is compiled.

EX PROGX; AFTER PROGZ

holds the execution of PROGX until PROGZ finishes executing.

See also:

RA (Remove After-Linkage) and RS (Remove Job From Schedule) commands.

AJ (DISPLAY ACTIVE JOBS)

Keyboard Input Message

Access Level: 0

The AJ command lists jobs that are currently executing. You can list a subset of the active jobs by using syntax options of the command.

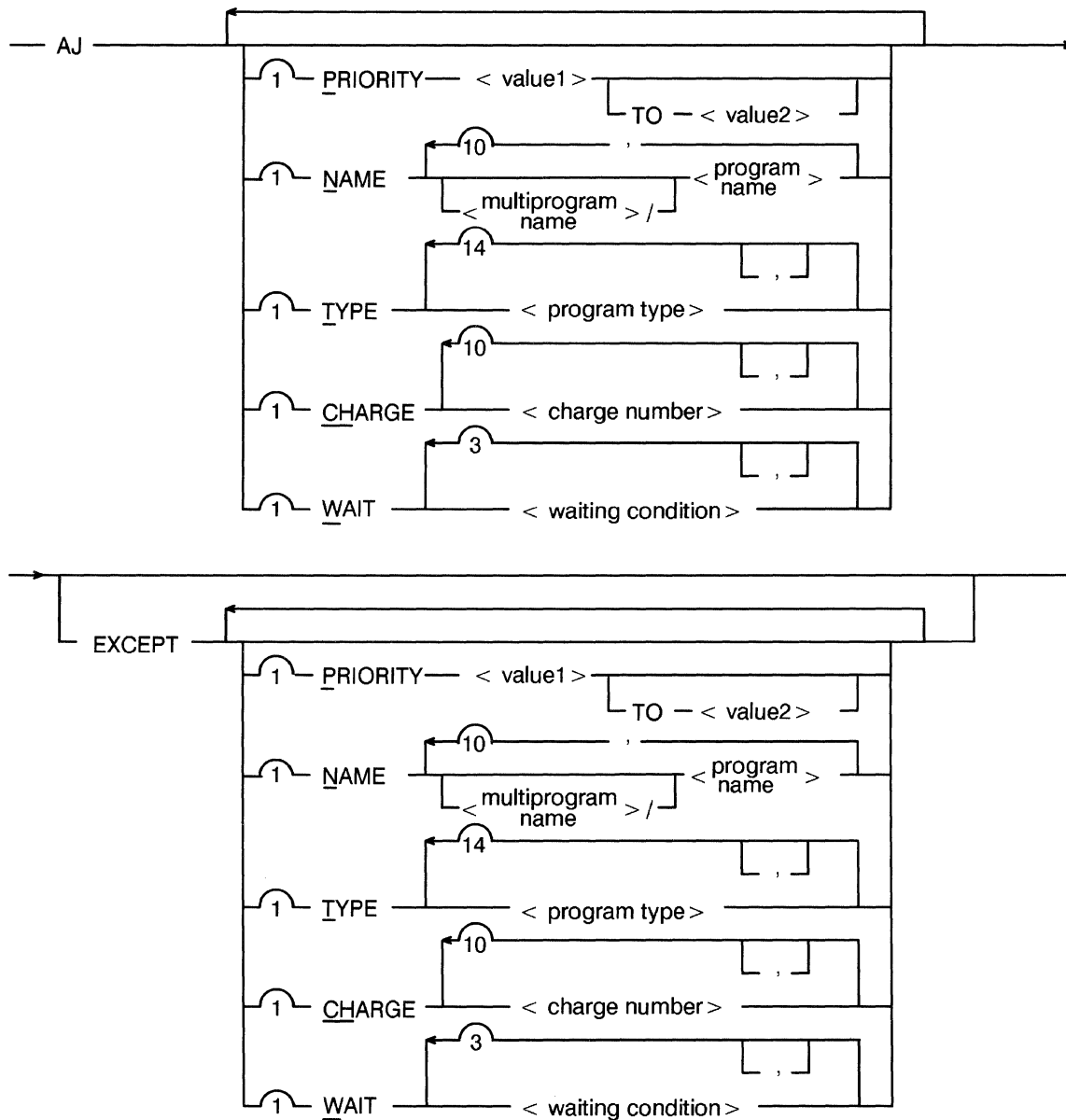


Figure 2-5. AJ Command Syntax

PRIORITY <value1> TO <value2>

Selects jobs based on processor priority. You have the option of selecting a single priority or a range of priorities. <value1> and the optional <value2> must be integers between 1 to 9. <value1> must be less than or equal to <value2>. The abbreviation P is accepted for PRIORITY.

NAME <multiprogram name> / <program name>

Selects jobs based on the program name and (optionally) on the multiprogram name. Separate the multiprogram name from the program name with a slash (/). You can enter a maximum of 10 name selections, separated by commas. The abbreviation N is accepted for NAME.

You can also select jobs based on part of a name by using the equal sign as a masking character. Use one equal sign for each unknown character position in the name. For example,

AJ NAME DMP=== EXCEPT NAME DMPOUT

displays all programs in the mix beginning with the three characters DMP, but does not display the program DMPOUT.

TYPE <program type>

Selects jobs based on the type of program. The abbreviation T is accepted for TYPE. You can enter a maximum of 14 type selections. The available program types are:

EXECUTE	Program is executing
COMPILER	Program generates object code
DMPALL	Program is DMPALL intrinsic
LOADMP	Program is LOADMP intrinsic
PACKUP	Program is PACKUP intrinsic
DSKOUT	Program is DSKOUT intrinsic or diskpack SQUASH intrinsic
MCS	Program is Message Control System
TIMESHARE	Program is executed as time sharing task
SHAREHNDL	Program is a time sharing handler program (for example, CANDE)
HANDLERS	Program is any type of handler (BNA, WFL, time sharing)
COMPSHAR	Program that generates object code, executed as a timesharing task
DMSPROG	Program is a DMSII control program (for example, DBP)
WFLHNDL	Program is a Work Flow Language (WFL) handler
BNAHNDL	Program is a BNA network architecture handler
COPY	Program is SYSTEM/COPY intrinsic
SORT	Program is SORT: or SORT. intrinsic
UTILITY	Program is any type of intrinsic (DMPALL, LOADMP, PACKUP, DSKOUT, SYSTEM/COPY)

■ CHARGE <charge number>

Selects jobs based on the charge number the job is executed under. You can enter as many as 10 charge number selections and 10 charge number exceptions. The abbreviation CH is accepted for CHARGE.

You can also select jobs based on part of a charge number by masking the charge number by using the equal sign as a masking character. Use one equal sign for each unknown character position. For example,

AJ CH 66===== EXCEPT CH 660001, 669999

displays jobs executed under a charge number beginning with the characters 66, but does not display jobs executed under charge numbers 660001 or 669999.

■ WAIT <waiting condition>

Selects jobs that are waiting for a specific condition to be satisfied (for example, waiting hardware, waiting memory and so forth). You can enter as many as 3 waiting conditions and 3 waiting condition exceptions. The abbreviation W is accepted for WAIT.

The available types of waiting conditions are:

<u>SPACE</u>	Waiting for memory, disk space or diskpack space
<u>DEVICE</u>	Waiting for hardware device to become available
<u>OPERATOR</u>	Waiting for operator to perform an action
<u>EVENT</u>	Waiting for a software event to be signalled (for example, STOQUE, CRCR and so forth)

■ EXCEPT

All text following the keyword EXCEPT is treated as exceptions to the selections entered previously. All of the exceptions must be grouped together after the keyword EXCEPT; you cannot enter a selection and an exception followed by another selection.

Details

The AJ command corresponds to the AD (Specify ODT Screen Parameters) command <table> entry AM for ODT display. AM displays the task number, processor priority, memory priority, and the amount of memory required for each job active in the mix.

If there are no active jobs, a "NULL MIX" response will be displayed.

See also:

AD (Specify ODT/OCS Screen Parameters); MX (Display Jobs In Mix); WJ (Display Waiting Jobs); and WS (Display Jobs In Schedule) commands.

ALLOCATE (Create Disk Space And Directory Entries)

MCP Control Instruction

Access Level: 8

ALLOCATE creates disk directory entries for files that can be loaded using the ALLOCATE feature of SYSTEM/COPY, the file transfer utility.

A number of optional elements permit considerable flexibility in the assignment of disk space for the file. You can also assign security attributes for the file.

The syntax for this command is identical to that for the ALLOCATE record of the system configuration file. See *V Series MCP/VS System Software Operations Guide Volume 1: Installation* for more information.

You cannot use more than one ALLOCATE command for each file-id.

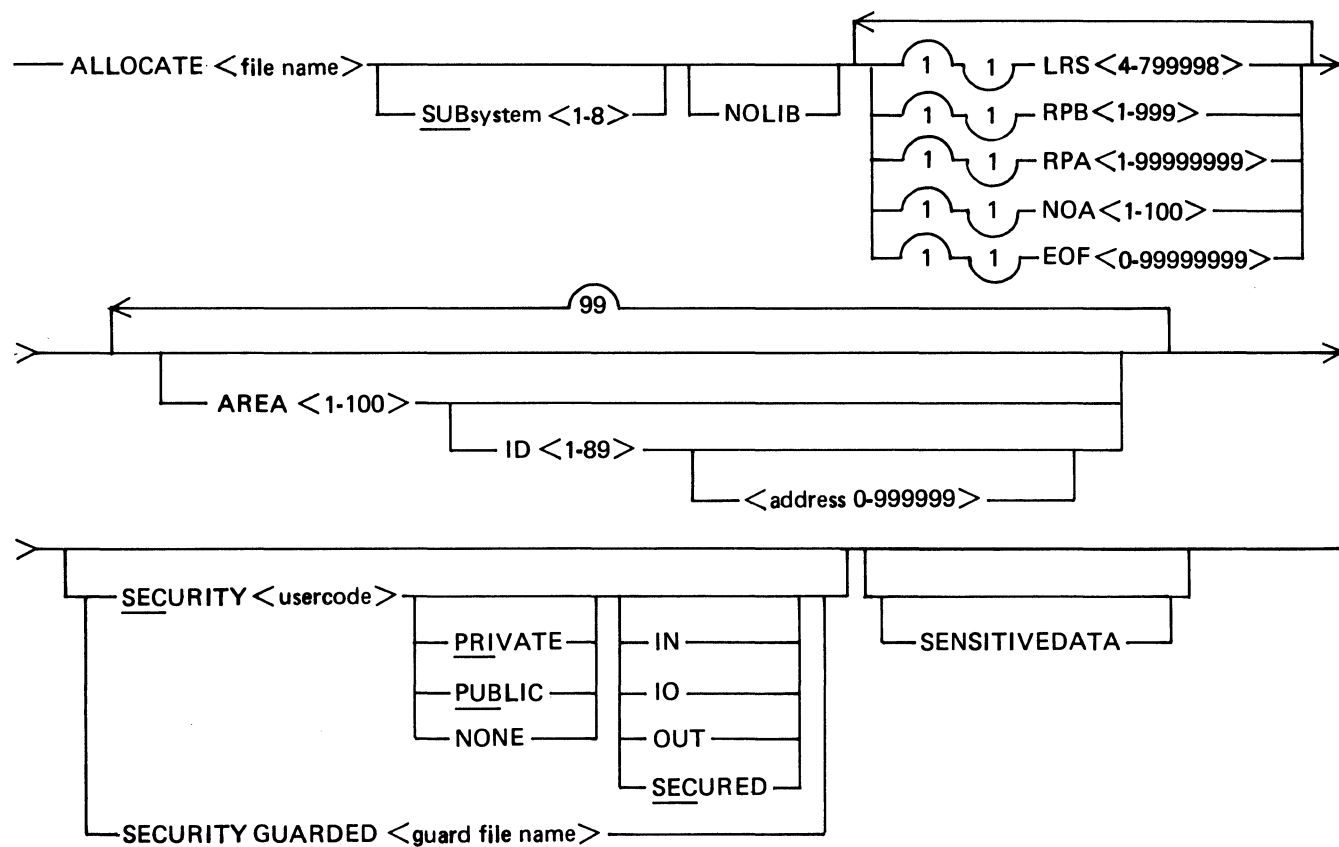


Figure 2-6. ALLOCATE Command Syntax.

<file name>

indicates the name of the file.

SUBsystem <1-8>

allocates the file to a specific logical disk subsystem. Logical subsystems numbers range from 1 to 8.

If you omit this option, the file is allocated to the default logical subsystem.

NOLIB

marks the file NO LIBRARY MAINTENANCE to prevent unwanted removal or changing of the file.

LRS <4 - 79998>

indicates the logical record size in digits. The value can range from 4 to 79998 digits and must be an even number.

RPB <1 - 999>

is the number of records per block. The size of a block (the number of records per block times the logical record size) is restricted only by the amount of available memory. The value can range between 1 and 999.

RPA <1 - 99999999>

is the number of records per area. The value can range from 1 to 99,999,999.

NOA <1 - 100>

indicates the file's number of areas. The maximum file size (the number of areas times the number of records per area) cannot exceed 99,999,999. The value can range from 1 to 100.

EOF <0 - 99999999>

indicates the end of file pointer. The value must fall within one off the areas allocated through an AREA specification or it must be zero.

AREA <1-100>

assigns areas of the file to specific disk devices and addresses. If you specify AREA, the file can be changed or removed through system commands, but it cannot be programmatically closed with a PURGE or with a REMOVE. Files with AREA specifications that include actual addresses are not squashed.

Allocating a multiple-area file requires all AREA allocations to be on one ALLOCATE command.

The AREA specifications are optional if the EOF value is 0 and are used to cause pre-allocation of disk space for the file. You can request a specific disk address, a specific ID, or allow the MCP to perform the allocation (according to any DISK record constraints).

ID <1 - 89>

is the disk device the area of the file is allocated to. If an address is used with this option, the allocated areas start at that address.

<address 0 - 999999>

indicates that the file or sector will begin at a specific address on the designated disk device.

SECURITY <usercode>

allows access to this file only when the specified usercode is given.

PRIVATE

lets only the creator access the file. This is the default security setting for ALLOCATED files.

PUBLIC

allows any user to access the file.

NONE

means that the file has no security assignment.

IN

allows data to be read from the file but not written to it.

IO

indicates that data can be read and written to the file.

OUT

indicates that data can be written to the file but not read from it.

SECURED

means that the file is a program and can only be executed. Data cannot be read from or written to this file.

SECURITY GUARDED <guard file name>

specifies that a special file must be accessed before this file can be accessed. The name of the guard file is <guard file name>. The guard file resides on disk.

SENSITIVEDATA

indicates that the file contains sensitive data. When the file is removed, its data will be replaced with random data.

Example

ALLOCATE COBTST LRS 200 RPB 1 RPA 500 NOA 20 EOF 0 SECURITY 8196 PUBLIC IO

allocates disk space for the file called COBTST. COBTST's physical and logical characteristics include a logical record size of 200, with 1 record per block. COBTST has a maximum of 20 areas and 500 records per area. Any user can access COBTST. Data can be read and written to the file.

ALLOCATE FILEZ LRS 200 RPB 100 NOA 2 EOF 200 RPA 100 AREA 1 ID 8 100000 AREA 2 ID 8 100100

allocates disk space for the file FILEZ. FILEZ's physical and logical characteristics include a logical record size of 200, with 100 records per block. FILEZ has 2 areas and 100 records per area. Area 1 is assigned to disk id 8 beginning at sector 100000; area 2 beginning at sector 100100.

See also:

COPY (Invoke File Transfer Utility) command.

The **ALLOCATE** record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

ALTER (CHANGE SYSTEM PARAMETERS)

MCP Control Instruction

Access Level: 8

The ALTER command lets you change several parameters that deal with the operating system, backup files, pack labels, and individual files as follows:

Operating System: You can set or change the maximum number of jobs allowed in the mix and schedule, the maximum number of system tasks, and the name of the MCP code file to be used. These parameters are set while the system is running, but do not take effect until the system is halt/loaded. This command can also change the MCP code file to be used the next time the system is halt/loaded.

Backup Files: ALTER changes the primary and secondary backup family names where printer backup files are directed by default. You can control the number of records per block and records per area of your printer backup files. You can change the next backup number assigned to printer or punch backup files. You can also set the lowest number used by the operating system when it searches for printer or punch backup files.

Pack Labels: You can set or change the pack family name, owner ID, and serial number.

Individual Files: You can set or change the LIBMAINT status of a file (whether the file can be changed or removed) and the SQUASH status of a file (whether it can be moved by the MCP intrinsic SQUASH).

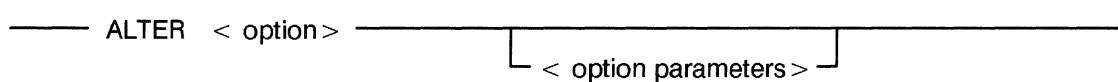


Figure 2-7. ALTER Command Syntax

<option>

indicates the option you want to alter and can be one of the options listed on the following pages.

BACKUP

changes the families where backup files are directed by default, the next number assigned to printer and punch backup files, and the lowest number that the operating system uses when searching for printer and punch backup files. Also changes the number of records per area and records per block of printer backup files.

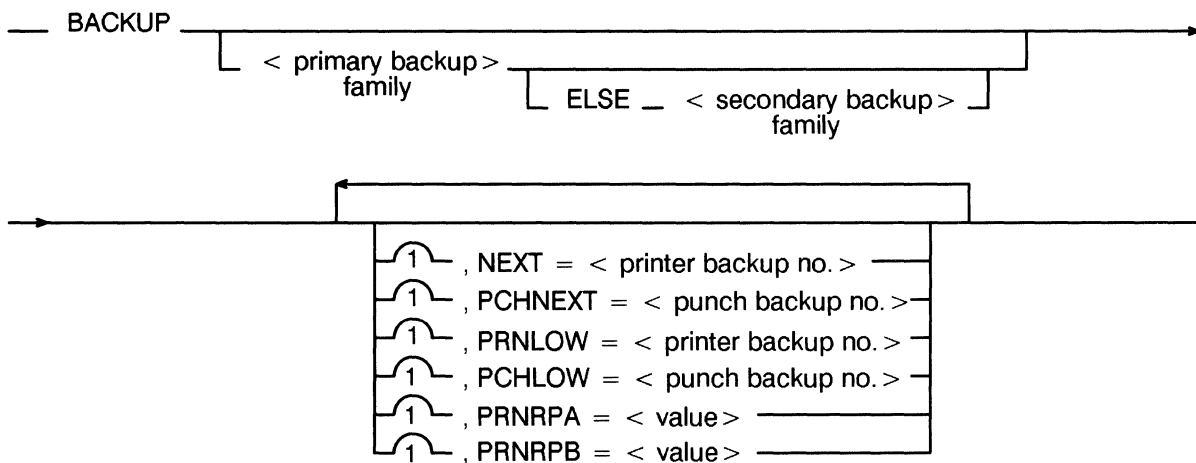


Figure 2-8. ALTER BACKUP Command Syntax

<primary backup family>

the family name of the diskpack family where the operating system stores printer backup files by default.

<secondary backup family>

the family name of the diskpack family where the operating system stores printer backup files if the primary backup family does not have enough space or is unavailable. If no secondary backup family is declared, the operating system stores backup files on any available system resource diskpack.

NEXT = <printer backup number>

sets the number that is assigned to the next printer backup file generated on the system. You cannot use 0; 1 is the lowest possible number.

PCHNEXT = <punch backup number>

sets the number that is assigned to the next punch backup file generated on the system. You cannot use 0; 1 is the lowest possible number.

PRNLOW = <printer backup number>

sets the lowest number that operating system intrinsics (for example, BFRF or PBDPRN) use when they search for printer backup files. You cannot use 0; 1 is the lowest possible number.

■ PCHLOW = <punch backup number>

sets the lowest number that operating system intrinsics (for example, BFRF or PBDPRN) use when they search for punch backup files. You cannot use 0; 1 is the lowest possible number.

■ PRNRPA = <value>

sets the number of records in each area of printer backup files. This value must be a multiple of 10 and a multiple of the PRNRPB value. If it is not a multiple of PRNRPB, the operating system will adjust it upward to the nearest multiple. The default value is 3150.

■ PRNPRB = <value>

sets the number of records in each block of printer backup file. This value must be a multiple of 10. The default value is 10.

EXCHANGE

adds or removes a channel from an exchange.

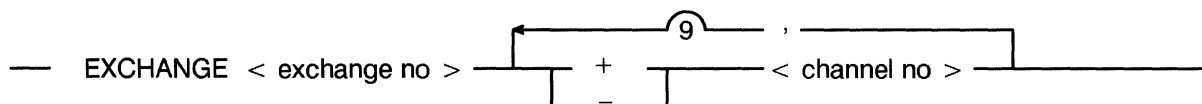


Figure 2-9. ALTER EXCHANGE Command Syntax

<exchange no>

is the number of the exchange from which you are adding or deleting a channel.

<channel no>

is the number of the channel which you are adding or deleting.

FILE

changes the LIBMAINT status and SQUASH status of a file.

If the LIBMAINT status is OFF, the file cannot be changed (using the CHANGE (Change File ID) command), written to, or removed.

SQUASH refers to consolidation of areas on a designated diskpack. If the SQUASH status is OFF, the file is *not* moved as a result of theSQP (Squash Diskpack) command.

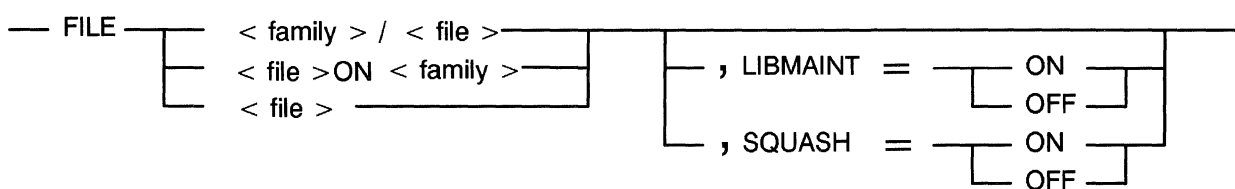


Figure 2-10. ALTER FILE Command Syntax

<file>

is the name of the file. FILE <file> indicates a disk file.

<family>

is the pack family name, if applicable. You could use FILE <file> ON DISK to indicate a disk file.

LIBMAINT

switches the LIBMAINT status on or off.

SQUASH

switches the SQUASH status on or off.

PACK

alters information about a pack device. However, the pack device must be saved (using the SV (Save A Peripheral Unit) command) before you can perform this operation.

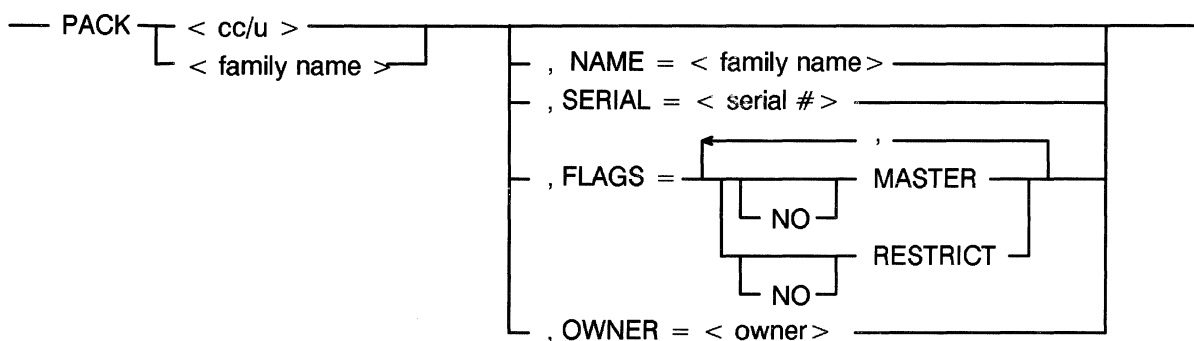


Figure 2-11. ALTER PACK Command Syntax

<cc/u>

<family name>

specifies the channel number/unit number (cc/u) of the pack device for which you want to alter information, or specifies the family name of a single-pack family.

<family name>

indicates a new pack family name.

<serial #>

indicates a new six-digit pack serial number.

FLAGS

switches on and off the flag to indicate that this pack device is the master pack for the pack family. Also switches on and off the flag to indicate that this pack is restricted.

<owner>

changes the name of the owner.

SYSTEM

changes the current MCP file or any of the special cold start constants. In order to use this option, you must first use the ALTER FILE option to set the LIBMAINT status of the new MCP file to OFF.

Note that multiple MCPs in a loosely coupled shared environment must be boot-compatible.

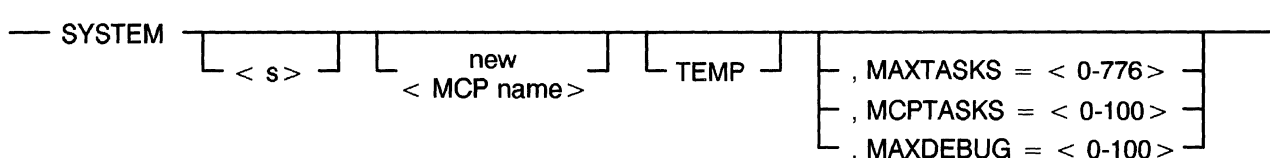


Figure 2-12. ALTER SYSTEM Command Syntax

<s>

is an optional system number and indicates in a multi-system shared configuration that only the specified system is to be affected.

If <s> is omitted, the MCP common to all systems is changed, otherwise the MCP for system <s> is changed.

<new MCP name>

indicates the new MCP file. The file must be on disk ID 01 and have LIBMAINT OFF (see the ALTER FILE option).

TEMP

causes the new MCP file to be loaded after the next halt/load. The new MCP file will run until there is another halt/load; the original MCP file will then be reloaded automatically.

MAXTASKS = <0-776>

alters the maximum number of tasks allowed in the mix and schedule.

MCPTASKS = <0-100>

alters the maximum number of independent runners (IRs) allowed on the system. Independent runners are special tasks initiated by the operating system to perform services for user tasks or for the operating system itself.

MAXDEBUG = <0-100>

alters the maximum number of debug sessions allowed.

Examples

ALTER BACKUP BKUP, NEXT = 0100

changes the default backup family to BKUP, and assigns 0100 as the next backup file number.

■ ALTER EXCHANGE 4 + 14

adds channel 14 to exchange number 4.

ALTER FILE NEWMBB, LIBM = OFF

switches the LIBMAINT status of the file NEWMBB off.

ALTER PACK 4/14, NAME = NEWPAK, SERIAL = 123456

changes the name and serial number of a pack device with channel number 4/unit number 14.

COPY MCPVS AS NEWMCP TO DISK (ID 01)

ALTER FILE NEWMCP, LIBM = OFF

ALTER SYSTEM NEWMCP, MAXTASKS = 200

changes the current MCP file for all systems to the file NEWMCP, and establishes a MAXTASKS parameter of 200.

COPY MCPVS TO DISK (ID 01)

ALTER FILE MCPVS, LIBM = OFF

ALTER SYSTEM 2 MCPVS

changes the MCP for system 2 to use MCPVS on disk. If all four systems were using a common MCP before this, the other systems will continue to use the common MCP, while system 2 uses a special MCP.

See also:

SHOW (Display System Information) command.

AT (Pass Message To Remote Host)

Keyboard Input Message

Access Level: 0

The local host recognizes the AT command as an ODT input message that is destined for a remote host. The local host does not check the message, but passes it to Host Services for transmission to the remote host. The remote host acts upon the message as though it were received from a local ODT subsystem and returns any responses to the originating host.

—— AT <hostname> <input message> ——

Figure 2-13. AT Command Syntax

<host name>

specifies the remote host that will receive and act upon the message.

<input message>

is any valid command or inquiry for the remote host.

Commands entered with AT are processed by the BNA program BNAODT. Consequently, use of the AT command requires the presence of BNA on the system. If AT is used and BNA is not running, the MCP returns the following message:

NOT IN NETWORK MODE

Example

AT SYSTEM2 DQ 30

causes a DQ command to be passed to a remote host named SYSTEM2.

See Also:

AT (Initiate BNA Job Transfer) MCP Control Instruction.

AT (Initiate BNA Job Transfer)

MCP Control Instruction

Access Level: 0

AT directs a system command to a remote host.

The AT command initiates a BNA job transfer dialogue (the transfer of job files) between the local host and a specified remote host.

Various commands can be included in this job transfer. Any number of jobs can be transferred to the same host in one job transfer process, and multiple jobs can execute simultaneously.

This command is valid only from an actual or pseudo card deck.

—— AT <hostname> ———|

Figure 2-14. AT Command Syntax

<hostname>

indicates the remote host to which the job is transferred.

Details

Each record following "?AT" in the pseudo deck is checked to determine if it is a control record, recognizable by a "?" in the first character position.

Each control record is further checked for an "END" command indicating the end of a job. Each data record or control record is transferred to the remote host until an ENDAT record terminates the dialogue. Each record deck starts with a ?AT <hostname> record and ends with a ?ENDAT record. (If the deck is from a pseudo reader, the ?ENDAT record is optional.)

Commands can be concatenated on one record separated by semicolons. There may or may not be commands following the hostname on the first record. If a command is concatenated with AT, the semicolon is optional.

An ?END control record indicates the end of a job and is optional for the last job in the file. MCP control instructions that can start an execution are: ADD, CHECK, COMPILE, COPY, COPYADD, DEBUG, DIR, DUMP, EXECUTE, GENERATE, LOAD, MOVE, PERFORM, and UNLOAD.

The following keyboard input messages can also start an execution: CV, ID, KA, KP, KS, PB, PC, PM, PP, or PS. These commands must be preceded by the word SPO. For example, ?AT HOSTA SPO CV5 is a valid SPO execution command.

One copy of BNAJXR enters the mix when a job is initiated with an AT command through the card reader or a pseudo card file. BNAJXR handles all commands entered that are preceded by AT, until an ENDAT is encountered. When ENDAT is encountered, BNAJXR terminates. Another AT command causes initiation of another copy of BNAJXR.

Commands preceded by the AT (Pass Message To Remote Host) keyboard input message are processed entirely by BNAODT. They do not cause initiation of BNAJXR.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

The following are examples of job files:

Example 1:

?AT YOURHOST PFM LIST1 INPUT CARD	% Beginning of Job 1
?DATA INPUT	% Control record
	%
[DATA RECORDS]	% Data records
	%
?END	% End of Job 1
?EX A ; DATA INPUT	% Beginning of Job 2
	%
[DATA RECORDS]	% Data records
?END	% End of Job 2
?ENDAT	% End of the file
	% ENDAT is optional

Example 2:

?AT SGHOST	% No command on first
	% record
?COMPILE PGMA COBOL9 LIB	% Beginning of Job 1
?DATA CARD	% Control record
<data records>	%
?ENDAT	%

See Also:

AT (Pass Message to Remote Host) keyboard input message and ENDAT (End Job Transfer) command.

AX (Respond To An ACCEPT)

Keyboard Input Message

Access Level: 0

The AX keyboard input message enters data into a program in response to an ACCEPT message.

— <mix no> AX <data> —

Figure 2-15. AX Command Syntax

<mix no>

is the mix number of the program requiring the data.

<data>

can be text up to 60 characters in length.

Alphanumeric Data: Data containing alphabetic characters, the digits 0 through 9, and spaces, can be entered immediately following the AX. The data begins with the first non-blank character following the AX and must end with a period or ETX character. Detection of any character other than those previously described will cause a syntax error. The data can be enclosed in quotes, in which case the terminating period or ETX character is not required.

Non-Alphanumeric Data: Data that contains non-alphanumeric characters or has leading spaces must have the entire string of characters enclosed within quotation marks.

Details

ACCEPT statements in a program cause the operating program to halt and wait for the appropriate data to be entered on the ODT. The ACCEPT function is available for BPL, COBOL ANSI-74, COBOL ANSI-68, Fortran, Pascal, and Work Flow Language (WFL) programs.

If the length of the data (including trailing spaces where quotes are not used) exceeds the size of the program's input buffer, the text is ignored and an error message is displayed.

If the text is shorter than the input buffer, the text is terminated with an ETX character in the program's ACCEPT area. The contents of any characters in the input buffer following the ETX are undefined.

An AX command can be entered for a program before the display of the ACCEPT message. If this is done, the MCP saves the data until the program issues the ACCEPT. In this case, if the length of the data entered exceeds the size of the program's input buffer, the data is truncated and no error message is displayed.

Examples

2 AX YES

responds to an ACCEPT message from the program associated with mix number 002 with the alphanumeric data YES.

15 AX "032080"

responds to an ACCEPT message from the program associated with mix number 015 with the non-alphanumeric data "032080".

See Also:

AX (Pre-Respond To An ACCEPT) MCP control instruction.

The ACCEPT function in the various programming language manuals.

AX (Pre-Respond To An ACCEPT)

MCP Control Instruction

Access Level: 1

The AX MCP control instruction places data into an ACCEPT queue for a program prior to its execution.

AX <data>

Figure 2-16. AX Command Syntax

<data>

can be text up to 60 characters in length.

Alphanumeric Data: Data containing alphabetic characters, the digits 0 through 9, and spaces, can be entered immediately following the AX. The data begins with the first non-blank character following the AX and must end with a period or ETX character. Detection of any character other than those previously described will cause a syntax error. The data can be enclosed in quotes, in which case the terminating period or ETX character is not required.

Non-Alphanumeric Data: Data that contains non-alphanumeric characters or has leading spaces must have the entire string of characters enclosed within quotation marks.

NOTE

AX can be intermixed with other job control commands, such as INSERT (Insert Data Into Program). However, each AX instruction must be terminated with a semicolon (;). No characters other than spaces can separate the final quote from the terminating semicolon.

Details

ACCEPT statements in a program cause the operating program to halt and wait for the appropriate data to be entered on the ODT. The ACCEPT function is available for BPL, COBOL ANSI-74, COBOL ANSI-68, FORTRAN, Pascal, RPG, and Work Flow Language (WFL) programs.

If the length of the data (including trailing spaces where quotes are not used) exceeds the size of the program's input buffer, the text is ignored.

If the text is shorter than the input buffer, the text is terminated with an ETX character in the program's ACCEPT area. The contents of any characters in the input buffer following the ETX are undefined.

Examples

EX COPY1; AX "COPY AAAAjm AS ZZZZjm"; AX "END"

starts the program COPY1 and enters into the ACCEPT queue the data "COPY AAAAjm AS ZZZZjm" and then the data "END".

COMPILE PROG1 COBOL9; AX "SET DEBUG"; INSERT 32 2 UN 66

compiles PROG1 with the COBOL9 compiler and enters into the ACCEPT queue the data "SET DEBUG", and inserts into the program at address 32 the data string 66, which is 2 digits of unsigned numeric data.

See Also:

AX (Respond To An ACCEPT) keyboard input message.

The ACCEPT function in the various programming language manuals.

BCL (Mark Beginning Of Card Data File)

MCP Control Instruction

Access Level: 0

The BCL command indicates the beginning of a punched card data file.

Whenever the BCL command is encountered in the card reader, it causes an End-Of-File (EOF) condition on any previous file and the unit is marked by the MCP as containing the new file.

This command is valid only from an actual or a pseudo card deck.

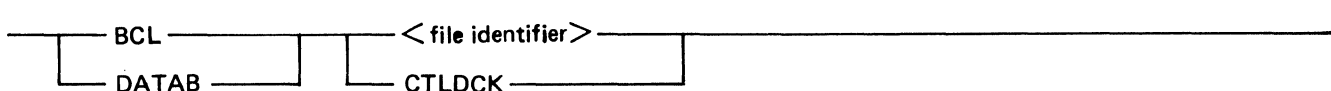


Figure 2-17. BCL Command Syntax

BCL
DATAB

are synonymous.

The word BCL specifies that subsequent input is in BCL format. The BCL language is a coded character set in which each character is represented by a six-bit code.

<file identifier>

is the name of the file to be read.

Any information after the <file identifier> on the same record is ignored.

CTLDCK

is a reserved file identifier that allows valid control records to be included in the file. CTLDCK is the name of the physical card deck used by the LDCNTL utility to build a pseudo card deck.

NOTE

If the <file identifier> is CTLDCK, no EOF action is taken until the ?ENDCTL control instruction is encountered. This is primarily intended for use by the utility program LDCNTL for building pseudo card decks.

Example

?BCL CARDS

specifies the beginning of a punched card data file called CARDS.

See also:

BINARY (Read Binary Card Deck); DATA/DATAB (Mark Beginning Of Card Data File); EBCDIC (Mark Beginning Of Card Data File); and END/ENDCTL (Mark End of Card File or Control Instructions) commands.

BEGINUSER (Mark Beginning Of Job Stream)

MCP Control Instruction

Access Level: 0

The BEGINUSER command is used for system security and delimits the beginning of a job stream and assigns a <usercode>/<password>/<charge no> access combination to all function requests in that job stream.

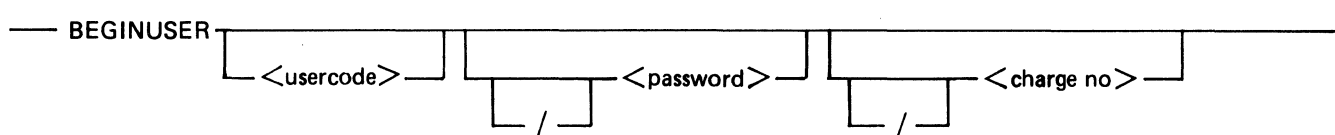


Figure 2-18. BEGINUSER Command Syntax

<usercode>

refers to a code that identifies a user to the system.

<password>

refers a group of characters by which the system determines the legality of a user.

<charge no>

refers to a number used to assign charges for system use.

NOTE

The <usercode>, <password>, and <charge number> are all optional. Slashes are required only if leading parameters are omitted (for example, BEGINUSER //55943).

Details

The BEGINUSER instruction can be entered whether or not the device from which it is entered is logged in.

The access combination specified in the BEGINUSER command is assigned to all functions initiated from the device until that combination is terminated by a matching ENDUSER (Mark End Of Job Stream) command with the same access combination. Alternatively, the specified access combination can be overridden for any single function initiated from the device by preceding the function request with a USER (Assign Access Code) command.

BEGINUSER and ENDUSER commands can be nested. However, the access combination specified on the ENDUSER instruction must always match the access combination of the last BEGINUSER instruction.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

BEGINUSER //92936

marks the beginning of a job stream, assigning a charge number of 92936 to all function requests in that job stream.

BEGINUSER 6720/PASSWORD/67073

marks the beginning of a job stream, assigning a usercode of 6720, password of PASSWORD, and charge number of 67073 to all function requests in that job stream.

See also:

ENDUSER (Mark End of Job Stream) and **USER** (Assign Access Code) commands.

The *V Series System Software Security Installation and Operations Guide* for information about <usercode>, <password>, and <charge no>.

BF, BFP (List Backup Files on Disk or Pack)

Keyboard Input Message

Access Level: 0

This command lists backup files according to the systems that created them, their backup numbers, their IDs, and their group IDs.

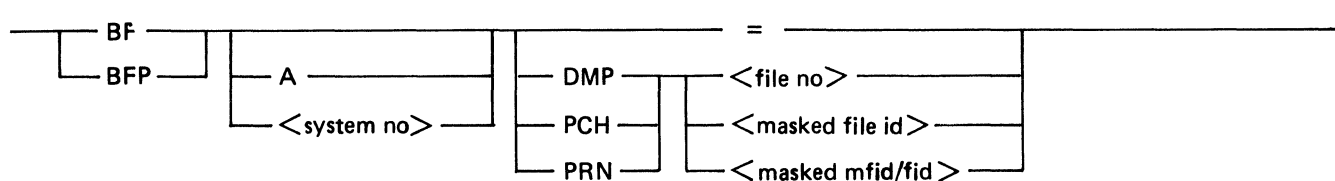


Figure 2-19. BF, BFP Command Syntax

BF
BFP

BF lists backup files on disk; BFP lists backup files on pack.

A
<system no>

The element A lists backup files produced by all systems. The <system no> lists backup files produced by a specified system.

If you omit this entry, the system will list only the files created on the system through which this request is made.

=
DMP
PCH
PRN

are acronyms that can be used to list various types of backup files: the equal sign lists all types of backup files; DMP lists dump files; PCH lists punch backup files; and PRN lists printer backup files.

Note that you cannot use BFP with dump files, as dump files are not written to pack. If you do, the message "****KBD IGNORED: INV SYNTAX" will be displayed.

<file no>
<masked file id>
<masked mfid/fid>

A <file no> lists only the name of the file with that number.

If you use a <masked file id>, the MCP lists all the backup files with that name. For PRN and PCH files, the name is a multifile-identifier/file-identifier or file identifier; for DMP files, the name is a program name that can be of the form multiprogram identifier/program identifier.

For printer and punch files, BF or BFP displays the total number of records in each file and the names of the programs that created them.

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

Examples

BF =

lists all the backup files on disk created by the system through which this request is made.

BF PRN PRINT

lists the printer backup files on disk named PRINT created by the system through which this command was issued.

BF DMP PROGA=

lists all the memory dump files with names beginning with PROGA.

BF PRN ABC=

lists all printer backup files on disk with names beginning with ABC.

See also:

BFRF (Search, List, Remove Backup Files) and RF (Remove Backup Files) commands.

BFRF (Search, List, Remove Backup Files)

Keyboard Input Message

Access Level: 0

The BFRF command performs the same functions as the BF (List Backup Files) and RF (Remove Backup Files) commands. However, instead of invoking the MCP, this command executes an intrinsic with additional file searching and listing features.

This program can list or list and remove backup files according to their types, numbers, and IDs. It can find the files according to name, date of creation, or backup file number. The files can be on disk or pack, families of packs, an individual system, or the entire system.

Another BFRF feature organizes the lists of backup files. It sorts lists alphabetically by name, sequentially by size, charge number, backup file number, or by the program that created the backup file.

Finally, another BFRF feature outputs those lists to a variety of media, including an ODT, a printer backup file on disk or pack, a line printer, or a disk file.

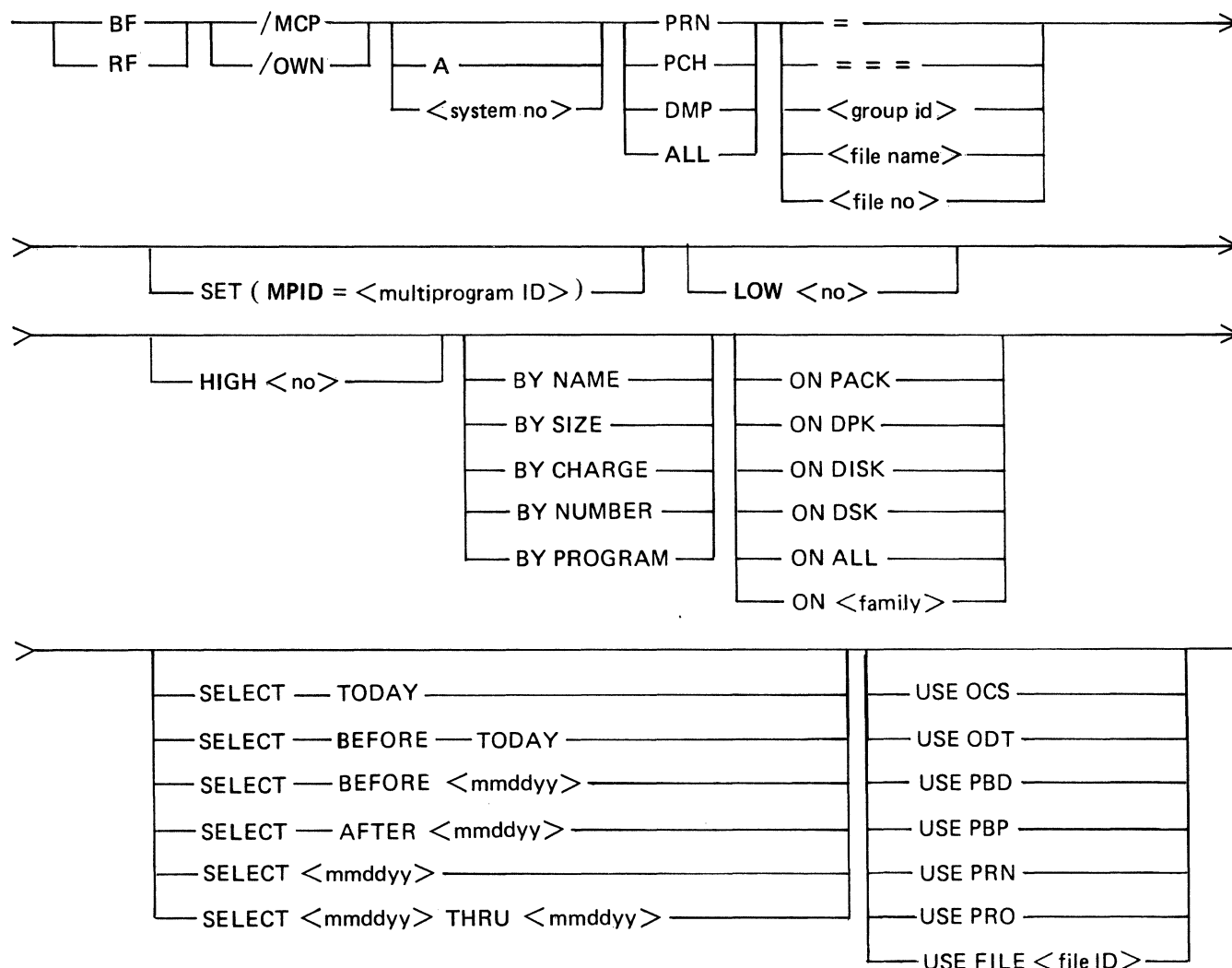


Figure 2-20. BFRF Command Syntax

You can declare the keywords SET, LOW, HIGH, BY, ON, SELECT, and USE in any order following the file identifier.

BF

RF

BF lists backup files. RF lists and removes backup files.

/MCP

/OWN

/MCP initiates the intrinsic bound to the MCP. /OWN initiates a freestanding version of BFRF, if one is on disk. If not, the MCP will display an error message that the program is not on disk.

A

<system no>

The element A initiates a search for files created on all systems.

The <system no> initiates a search for files created on the system with that number.

If the character A or a system number is not used, the search defaults to the system that initiates BFRF.

PRN

PCH

DMP

ALL

indicates the type of backup files to search for. PRN initiates a search for printer backup files; PCH for punched backup files; DMP for dump files; and ALL for all types of backup files.

=

===

<group ID>

<file no>

<file name>

indicate which files to list or remove.

Equal signs can be used as masking characters to indicate that all files of a designated type should be removed. RF requires exactly three equal signs, while BF can use one or more equal signs.

If you use <group ID>, BFRF lists or removes the files with matching names.

If you use <file no>, BFRF lists or removes that file only. Note that when listing or removing files by backup number, the MCP commands BF/BFP or RF/RFP are recommended. This avoids running a program just to list or remove a single file.

If you use <file name>, BFRF lists or removes backup files with that name. For PRN and PCH files, the name is a <group ID> (multifile identifier/ file identifier) or file identifier. For DMP files, the name is a program name that can be in the form multiprogram ID/program ID.

SET (MPID = <multiprogram ID>)

gives different executions of BFRF different names. For example, three BFRF operations in the mix simultaneously could be named MOE/BFRF, LARRY/BFRF, CURLY/BFRF. These names would appear on the Run log and make scheduling of other programs easier after a particular execution of BFRF.

LOW <no>

HIGH <no>

are limits that can be used to search only those backup files numbers within the specified range. LOW defines the low end and HIGH defines the high end of that range. Both the LOW number and the HIGH number can be up to four digits long. A fifth digit is automatically taken from the system number.

LOW and HIGH are both optional. You can use one without the other. If you do not specify the LOW end of the range, it defaults to one. If you do not specify the HIGH end of the range, it defaults to 9999.

BY NAME

BY SIZE

BY CHARGE

BY NUMBER

BY PROGRAM

are parameters that sort lists alphabetically by name or by the name of the programs that created them. You can also sort sequentially by size, by charge number, by backup file number or by the program name that created the backup file. If you do not use these options, BFRF lists the files in the same order in which they appear in the directory.

ON PACK

ON DPK

ON DISK

ON DSK

ON ALL

ON <family>

are parameters that narrow BFRF's search of a particular medium. ON <family> focuses the search to a particular pack family, which can be a restricted pack family. PACK and DPK focus the search on resource packs. BFRF searches all disk and unrestricted pack by default or when you specify ALL.

SELECT TODAY

SELECT BEFORE TODAY

SELECT BEFORE <mmddyy>

SELECT AFTER <mmddyy>

SELECT <mmddyy>

SELECT <mmddyy> THRU <mmddyy>

BFRF can list or remove files according to the dates that they were created. The <mmddyy> figure must be six characters. For example, the date 1/1/85 is written 010185; the date 9/14/82 is written 091482; and the date 2/2/84 is written 020284.

USE OCS

USE ODT

USE PBD

USE PBP

USE PRN

USE PRO

USE FILE <file ID>

are parameters that output the lists to several media: line printers, disk and pack printer backup files, or disk files. USE ODT is the default.

If the USE option specifies FILE, the information is not formatted for display but is returned in a predefined format. The program accessing the file can then use that data as it is or format it as required.

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

The attributes of the disk file created by the USE FILE option are:

FAMILY	=	DISK
RECORDSIZE	=	44 BYTES (per format below)
BLOCKSIZE	=	20 RECORDS
TITLE	=	<file ID>

The format of the disk file is:

```

01 BF-RECORD.
  03 BF-ID-FIELDS.
    05 BF-MULTI-FILE-ID          PICTURE X(06).
    05 BF-FILE-ID                PICTURE X(06).
    05 BF-CREATOR-MULTI-PROG-NAME PICTURE X(06).
    05 BF-CREATOR-PROG-NAME      PICTURE X(06).
    05 BF-FILE-TYPE              PICTURE 9(01)  COMP.
    05 BF-BK-NUMBER              PICTURE 9(05)  COMP.
  03 BF-STATUS-FIELDS-PRN        COMP.
    05 BF-CREATION-DATE-YYDDD     PICTURE 9(05).
    05 BF-CHARGE-NUMBER           PICTURE 9(06).
    05 BF-EOF-PB-PC              PICTURE 9(06).
    05 BF-QT-LINE-PB             PICTURE 9(06).
    05 BF-QT-FLAG-PB             PICTURE 9(03).
  03 BF-STATUS-DMP-BRK
    REDEFINES BF-STATUS-FIELDS-PRN COMP.
    05 FILLER                     PICTURE 9(05).
    CREATION DATE
    05 FILLER                     PICTURE 9(06).
    CHARGE NUMBER
    05 BF-EOF-DMP-BRK             PICTURE 9(05).
    05 BF-DC-DATE-DMP-BRK         PICTURE 9(06).
    05 BF-DC-TIME-DMP-BRK         PICTURE 9(04).
  03 BF-ATTRIBUTES               COMP.
    05 BF-MEDIA                  PICTURE 9(01).
    05 BF-NO-CHANNEL-12          PICTURE 9(01).
    05 BF-UNLABELED              PICTURE 9(01).
    05 BF-FORM                   PICTURE 9(01).
    05 BF-ATTRIBUTE-2            PICTURE 9(01).
    05 FILLER                    PICTURE 9(03).

```


Note that it is the user's responsibility to make the file ID unique.

If the BFRF search does not find any files and you specify the USE FILE option, the system still creates a disk file even though it contains no records.

If <multiprogram ID> is not specified, the system places the program identifier in the variable BF-CREATOR-MULTI-PROG-NAME (for sorting purposes) and makes the variable BF-CREATOR-PROG-NAME blank spaces.

The variable BF-FILE-TYPE defines the type of backup file as follows: 0=printer, 1=punch, and 2=dump. Some fields have no meaning when BF-BACKUP-TYPE is other than 1 (for example, BF-FORM and BF-NO-CHANNEL-12).

If the file has been stopped by the QT (Quit Program Operation) command, the BF-QT-LINE-PB and BF-QT-FLAG-PB fields will contain, respectively, the last record number printed and the three-hexadecimal QT flag.

The variable BF-ATTRIBUTE-2 is the medium on which the program was executing for dump files. Otherwise, this variable is zero. BF-MEDIA is the location of the backup file itself. The value of BF-ATTRIBUTE-2 is 0 for disk and 1 for pack. The value of BF-MEDIA is 0 for disk and 1 for pack.

Executing The BFRF Intrinsic

You can execute the BFRF intrinsic with the EXECUTE (Execute Library Program) command.

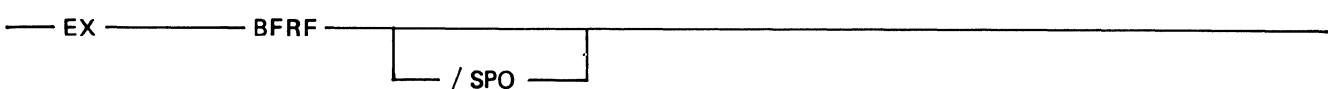


Figure 2-21. Executing BFRF With The EXECUTE Command

EX BFRF

executes the BFRF intrinsic.

When the MCP schedules BFRF from a BF/RF /MCP or /OWN command, the system provides a passfile containing the syntax entered for the program. When executed, the passfile does not exist and the program looks for a parameter file on disk ("BFPARM") to define the specifications to be used. BFPARM consists of 100-character records blocked 1 on disk can be file equated.

If the parameter file does not exist, the program will ask for input in the form of AX (Respond To An ACCEPT) command. Enter either "<mix no> AX or "<mix no> AX END" to terminate this form of input.

EX BFRF/SPO

also executes the BFRF intrinsic, but provides no check for a parameter file and requests input in the form of an AX (Respond To An ACCEPT) command. Enter either "<mix no> AX or "<mix no> AX END" to terminate this form of input.

Examples

RF /OWN PRN ===

initiates the freestanding version of BFRF to search on disk and unrestricted packs and remove any printer backup files created by the system through which this command is entered. By default, output the list of removed files to the ODT.

BF /MCP 1 ALL = ON DISK USE OCS

initiates the bound intrinsic BFRF to search on disk for all types of backup files created by system number 1. Output that list on the OCS.

BF /MCP 2 ALL === BY NAME USE ODT

initiates the MCP's bound intrinsic BFRF. It seeks backup files of all types created by system 2, sorts the list alphabetically by file name, and outputs that list to the ODT.

RF /OWN A DMP === BY CHARGE

initiates a freestanding program BFRF. It seeks and removes dump backup files created by all the systems in a multi-system shared configuration. Then it sorts the list of the removed files sequentially by charge number. Because no particular medium is specified, it outputs that list to the ODT.

BF /OWN A PRN FILE== LOW 1 HIGH 700

ON PACK SELECT TODAY BY NAME USE OCS

initiates a freestanding version of BFRF. It lists printer backup files on pack created by all the systems in a multi-system shared configuration. Of those files, it lists only those created today with backup numbers ranging between 1 and 700. Finally, it sorts the list alphabetically by file name and outputs it to the OCS.

See also:

BF, BFP (List Backup Files); (EXECUTE (Execute Library Command); QT (Quit Program Operation); and RF, RFP (Remove Backup Files) commands.

BINARY (Read Binary Card Deck)

MCP Control Instruction

Access Level: 0

The BINARY command reads a binary card deck.

The deck must be terminated by a binary end card to allow normal End-Of-File (EOF) operation.

Since the format of this card is difficult to keypunch, it is recommended that a binary punch file be opened and closed several times to produce a supply of these cards.

A binary card deck must come directly from the reader; it is not valid under LDCNTL, the system utility that creates pseudo decks from physical card decks.

—— BINARY <file name> ——|

Figure 2-22. BINARY Command Syntax

<file name>

specifies the name of the card deck to be read.

NOTE

The <file name> of CTLDCK is reserved and should not be used.

Example

EXECUTE A; BINARY ACARD

executes the library program called A, then reads a binary card deck called ACARD.

See also:

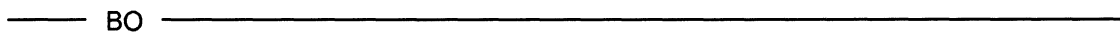
BCL (Mark Beginning of Card Data File); DATA/DATAB (Mark Beginning Of Card Data File); EBCDIC (Mark Beginning Of Card Data File); and END/ENDCTL (Mark End of Card File or Control Instructions) commands.

BO (BLACKOUT ACCESS CODE)

Keyboard Input Message

Access Level: 0

To keep access codes secret, you can use the BO command to prevent unauthorized persons from viewing a particular access code as it is entered during the log-in procedure.



```
_____ BO _____|
```

Figure 2-23. BO Command Syntax

Details

You invoke the blackout feature by entering the BO command before logging in. This command “blacks out” a line on the input device and leaves the carriage positioned at the beginning of the line.

See also:

LI (Log In) command.

BT (BEGIN TRAK)

Keyboard Input Message

Access Level: 8

The BT command selects a set of TRAK calls and causes the TRAK facility to begin tracking those calls. This command is primarily used for debugging the operating system.

TRAK is a real-time debugging tool that records various MCP data structures or events. TRAK calls (sometimes called TRAK "invocations") are embedded in the operating system at various points. Each TRAK call causes certain data to be stored in a portion of memory called the TRAK buffer. The TRAK buffer is included in the memory dump file and can be analyzed by DMPANL, the dump analysis utility.

When you select a set of TRAK calls using the BT command, that set of BT specifications remains in effect across halt/loads. The BT command can select TRAK calls based on:

- • TRAK code number (unique to each TRAK call)
- • The name of an operating system module
- • The kind of data that a TRAK call inserts in the TRAK buffer
- • The task that the operating system was executing for when the TRAK call was encountered

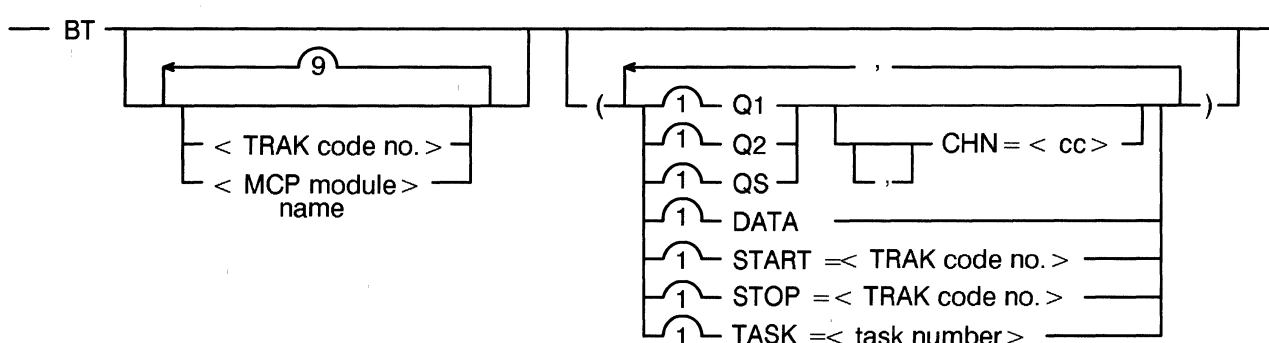


Figure 2-24. BT Command Syntax

■ <TRAK code number>

Selects TRAK calls based on the 4-digit code used to identify a TRAK call and its subsequent entry in the TRAK buffer. This code (and any text interpretation of it) are printed out by DMPANL in its analysis of the TRAK buffer. You can select a maximum of ten different TRAK codes.

■ <MCP module name>

Selects TRAK calls contained within a set of operating system module(s). You can select a maximum of ten different modules.

Q1
Q2
QS

These options select TRAK calls that insert copies of the I/O queue elements into the TRAK buffer. Q1 selects queue elements of I/Os to be initiated (fired). Q2 selects queue elements of completed I/Os. QS includes both types.

CHN = <cc>

Valid only with the Q1, Q2 or QS options. This option selects TRAK calls that insert I/O queue elements into the TRAK buffer and further selects only those occurrences of a TRAK call where the queue element is related to a specific channel number.

If the QS option is used and the operating system module IOMOD is selected, the CHN option can select I/O-related TRAK calls in IOMOD that relate to a specific channel (although IOMOD TRAK calls are not Q1 or Q2 TRAK calls).

DATA

Selects TRAK calls that insert non-standard TRAK data into the TRAK buffer. (Non-standard includes any data other than the I/O queue elements or the stack frames and parameters).

START = <TRAK code number>

Starts tracking selected TRAK calls only after the entered TRAK code number is encountered.

STOP = <TRAK code number>

Suspends tracking selected TRAK calls after the entered TRAK code number is encountered.

TASK = <task number>

Selects only those occurrences of TRAK calls that relate to the entered task number.

Details

The TRAK option can be enabled either through the USE TRAK record of the system configuration file or through the SO (Set System Options) command.

The tracking initiated by the BT command is terminated by using the ET (End TRAK) command.

Examples

BT IOMOD,DCP (TASK = 042)

begins tracking of the MCP modules IOMOD, and DCP; includes only TRAK calls relating to task 42.

BT 9310,1200,1234 (START 8500)

begins tracking of TRAK calls associated with TRAK codes 9300, 1200, and 1234 after TRAK code 8500 is encountered.

BT IOMOD (QS,CHN = 26)

begins tracking of all I/Os initiated and completed on channel 26.

See Also:

ET (End TRAK), PM (Print Memory Dump), SO (Set System Options) commands.

The USE TRAK and LIMIT TRAKBUFFER records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The DMPANL utility in *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

BUFFER (SET MCS RESULT QUEUE SIZE)

MCP Control Instruction

Access Level: 0

The BUFFER command sets the number of input messages a Message Control System (MCS) result queue can store.

Input this command when:

- initiating a MCS program that uses the Data Communications Processor (DCP) file as specified by <DCP file name>
- the DCP file as specified by <DCP file name> is closed and the DCP option is set.

Once the buffer size is set, that size remains in effect until it is changed by another BUFFER command or the DCP option is reset.

— BUFFER < DCP file name> = < integer> —

Figure 2-25. BUFFER Command Syntax

<DCP file name>

must match either the name defined in the "MCSNAME =" statement in the B 974 NDL or in the "MCS =" statement in the B 874 NDL. The name can be up to six characters long.

=

is optional and does not influence the command.

<integer>

indicates the number of messages the buffer holds. The number can be from 1 to 99. The default is 5.

Use <integer> carefully. A low value can cause unnecessary delays in receiving a message from the DCP. Large values, on the other hand, use large amounts of memory.

NOTE

Note that the memory size (in digits) of an MCS result queue in a DCP file equals <max record size in digits> * <no of messages in the result queue + 1> + 200.

Example

BUFFER OURDCP 15

allows the result queue in the DCP file called OURDCP to store up to 15 input messages.

See also:

The *V Series BNA Installation and Operations Guide* for more information.

CA (List Port Level Candidates)

Keyboard Input Message

Access Level: 0

The CA (CANDIDATE) command provides information about the current port level candidates, which are those ports and subports that are in the process of opening; they may be awaiting matching cooperators or be in some phase of the matching process.

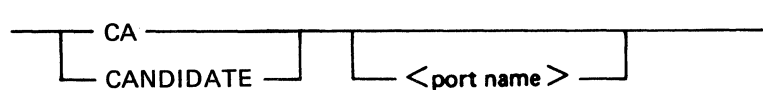


Figure 2-26. CANDIDATE Command Syntax

CA CANDIDATE

are synonymous and, if used without the <port name> option, return a list of all candidates with summary information about each.

<port name>

displays a list of only those candidates with this port name, and displays summary information about each.

The system response takes the format shown in Figure 2-27.

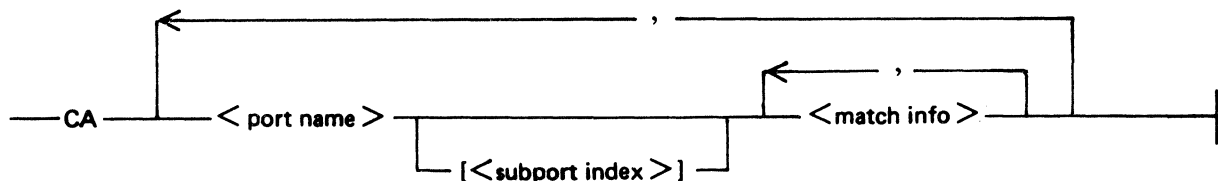


Figure 2-27. Response to CANDIDATE Command

[<subport index>]

is returned only for local candidates.

<match info>

contains a listing of the port's attributes, and provides the status of the port within the matching process.

Example

CA LIOPORT

displays information about LIOPORT. A sample response is as follows:

CANDIDATES: LIOPORT SUBPORT = 0001 FROM (???) DBG450/DISK /0004

STATUS = OFFER OUTSTANDING

SECURITY = PRIVATE YOURUC = boU

MYHOST = HOSTQ

YOURHOST = HOSTE

MYNAME = LIOINIT

YOURNAME = LIOCOOP

See also:

The V Series BNA Installation And Operations Guide.

CC (Enter Control Instructions)

Keyboard Input Message

Access Level: 0

The CC command supplies control instructions and text to the MCP through the ODT. The information following the letters CC is recognized in the same way as information following an invalid character in column 1 of a Control record.



Figure 2-28. CC Command Syntax

<control information>

is any valid MCP control instruction except DATA/DATAB (Mark Beginning Of Card Data File), LABEL1 (Change Tape Label Format), or SPO (Simulate SPO Through Card Reader). When entering control information, note the following:

- All control information including program parameter instructions must be entered on the same CC line.
- The control information must be separated from the initial letters CC by at least one blank space.
- A hyphen (-) cannot be used to continue an instruction to the next line.

Details

The CC message is not required prior to entering any control information. It is provided to maintain compatibility with previous versions of the MCP. For example, if a system running MCP/VS 2.0 is communicating with a system running MCP/VS 1.0, system commands must be prefaced with the characters CC if they are entered through BNA through an ODT and start a job.

While the CC command allows you to enter MCP control instructions, the SPO (Simulate SPO Through Card Reader) command provides capability for keyboard input messages to go to the MCP through a card reader as if they had been entered at the ODT.

Examples

CC REMOVE A=

allows you to enter a REMOVE (Delete Files) command to delete the files with the file id of A= from the disk directory.

CC COMPILE BOB WITH COBOL9 LIBRARY; PRIORITY 6

allows you to enter a COMPILE (Compile Program) command to compile a COBOL program called BOB and then save it as a permanent file. This job is assigned a memory, processing, I/O, and schedule priority of 6.

CC EXECUTE TESTY; MEM 400000; MEMDUMP

allows you to enter an EXECUTE (Execute Library Program) to execute a program called TESTY, specifying that the program needs 400000 digits of memory to execute. If the program terminates abnormally, a dump is requested.

CC CHANGE A=B=C= TO X=Y=Z=

allows you to enter a CHANGE (Change File ID) command to change the name of files on disk with a file id from A=B=C= to X=Y=Z=.

See also:

SPO (Simulate SPO Through Card Reader) command.

CD, CDP (List Inactive Pseudo Decks)

Keyboard Input Message

Access Level: 1

The CD command lists on the ODT the inactive pseudo card decks on disk; CDP lists the inactive decks on pack. The MCP responds with a list of the file-numbers and the contents of the first card image in the control decks.

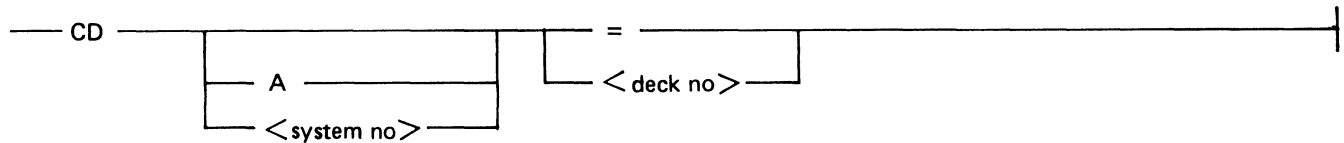


Figure 2-29. CD Command Syntax (Format 1 - Disk)

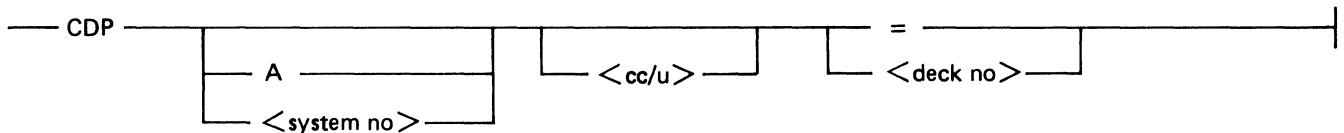


Figure 2-30. CDP Command Syntax (Format 2 - Pack)

Format 1

CD

lists the inactive pseudo card decks on disk.

A

indicates a system specification of all systems.

<system no>

can be specified in a multi-system shared configuration to obtain a list of the inactive pseudo decks created on the indicated system.

If you omit either A or <system no>, the response includes only those decks created on the system through which the request is made.

=

lists all decks of the indicated system.

<deck no>

causes the MCP response to include only the specified deck.

Pseudo decks are designated as #snnnn. The s represents the system number and is not required unless the deck was created by a system other than the one from which you are entering the command.. The nnnn is a four-digit number with leading zeros assumed.

Do not include the pound sign in the command syntax.

Format 2

CDP

lists the inactive pseudo card decks on pack.

<cc/u>

can be used only with CDP and causes files from a pack with the specified channel number/unit number to be displayed.

Examples

CD A =

displays a list of all the inactive pseudo card decks on all the systems.

CD 1 3

displays the inactive pseudo card deck number 3 on system number 1.

CDP 6/3 =

displays a list of all the inactive pseudo card decks from the diskpack on channel number 6/unit number 3.

See also:

DA, DAP (Deactivate Pseudo Reader); PD (Print Directory); RD, RDP (Remove Pseudo Decks); and RN (Activate Pseudo Reader) commands.

CHANGE (Change File ID)

MCP Control Instruction

Access Level: 5

The CHANGE command changes a disk or pack file id. If the SECURITY option (DFSC) is set and the file has a SECURITYTYPE other than PUBLIC, the change can be made only under the access code of the creator of the file or by a user with LIBMAINT privileges.

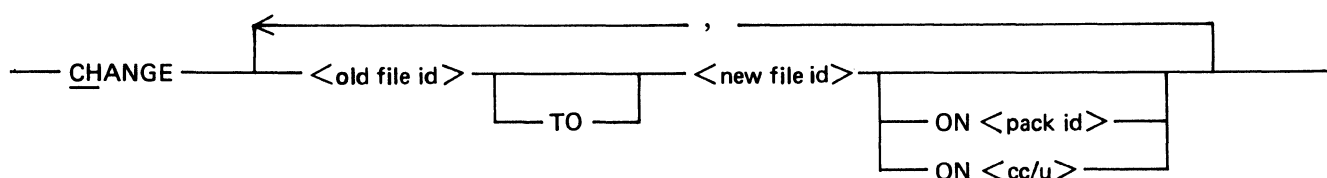


Figure 2-31. CHANGE Command Syntax

CHANGE
CH

are synonymous.

<old file id>
<new file id>

indicate the name of the file id you want to change and the file id that will replace it.

If the <old file id> is not masked and more than one BASE area exists with the specified name (a BASE area is the first area of a pack file), a DUP FILE message will be displayed.

If a masked <old file id> is specified and multiple files exist that match the <old file id>, only the first matching file encountered will be CHANGED. All subsequent matching files will get DUP FILE errors.

An attempt will be made to CHANGE all matching files. A file will not be CHANGED if an existing file has the same name. A DUP FILE error will be produced in this case.

If any specified files are IN USE or declared to have NO LIBMAINT, those files will not be CHANGED.

If <new file id> specifies a pseudo device, the files will be CHANGED to ascending-numbered back-up files of the appropriate type.

If a file is CHANGED to a pseudo-reader device and APCR is set, an attempt will be made to run the resulting pseudo decks. If you enter CH <filename> # or CH <file name> #= and APCR is set, All pseudo card decks that have not been activated through an RN, RNP (Activate Pseudo Reader) command previously will be activated.

If any continuation areas of CHANGE candidates cannot be located because of missing packs, only the present continuation areas will be CHANGED and the BASE area will be flagged as CHANGE IN PROGRESS. All further access to the file will be restricted, although another CHANGE command can be issued when the missing packs are made available to the system. The BASE area pack must still be present for all subsequent CHANGE attempts.

When all continuation areas have been CHANGED, the BASE area will be CHANGED. If for some reason the entire file is not available to complete the CHANGE, a REMOVE (Delete Files) command can be issued to remove any areas that have not yet been CHANGED. Areas that have been CHANGED will not be located by a REMOVE command even if present in the system. Eventually, option 3-type REMOVES will have to be performed to restore pack integrity.

Pseudo-Device File-Identifiers

Using the following identifiers, the file id of a file, or of a series of files, can be changed to a pseudo-device file identifier recognizable by the MCP:

File Identifier	Function
@	Creates a printer backup file.
*	Creates a punch backup file.
#	Creates a pseudo card deck file.
\$	Creates a memory dump file.

Each of the preceding files are assumed to be in the correct format to complete the CHANGE because a check is not made by the MCP.

File-group identifiers are allowed. Trailing zeros are not permitted on the pseudo-device file-identifiers.

ON <pack id>

If the ON clause is omitted or the <pack id> is DISK, the files to be changed are assumed to reside on disk (100-byte media). If the ON clause specifies a <pack id> of PACK, all system resource packs will be searched for the specified file group <old file id>.

If the pack-id is a valid pack, all packs with the specified name will be found. If no file BASE areas are found which match the <old file id>, a NO FILE error message will result.

ON <cc/u>

The cc/u (channel/unit identifier) must specify a pack. All files specified by <old file id> will be located. If no files are found on the specified cc/u (BASE areas only) matching the <old file id>, a NO FILE error will result.

A duplicate file check for the new id will be done across the entire pack family, not just the specified cc/u.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

CH FILEB TO *

changes the file id FILEB to the next available punch backup file id.

CH FILEC TO @00256

changes the file id FILEC to printer backup file 00256.

CH FILE= TO @

changes all files on disk beginning with FILE to the next available pseudo-device file identifier; therefore, the operator is notified of the CHANGE regardless of the setting of the MCP LIB option.

CHANGE PBD= TO @

causes the file identifiers of all files beginning with PBD to be changed to printer backup file identifiers.

CH A TO B, C D, E TO F, G= H=

changes file id as follows: A to B, C to D, E to F, file id's beginning with G to file id's beginning with H.

CHANGE ABDFIL TO XYZFIL ON 15/3

changes the file id of a file on a pack with channel number 15, unit number 3.

See also:

File (Equate File-Name With File-Identifier Or Medium) and REMOVE (Delete Files) command.

CHARGE (Insert Charge Number)

MCP Control Instruction

Access Level: 1

The CHARGE instruction inserts a user charge number into the MCP Run log (RLOG) records for the program with which it is used.

The CHARGE command can be used to assign a user-designated charge number to the execution of any program. If the CHRG option is set, all programs being scheduled for operation must have a charge number to be accepted by the MCP.

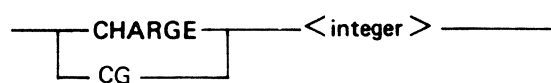


Figure 2-32. CHARGE Command Syntax

CG
CHARGE

are synonymous.

<integer>

must be a number not longer than six digits. If fewer than six digits are used, leading zeros are assumed. This number is carried in the RLOG file for subsequent time analysis or billing use.

Details

The ZIP facility allows user programs to send system commands, and the commands sent with the ZIP facility are acted on exactly as if they had been entered at the ODT. The ZIP facility is invoked by constructs in the various programming languages. When a program uses a ZIP statement to schedule another program for operation and fails to supply a charge number for the ZIPped program, the charge number of the program doing the ZIP will be used for the ZIPped program.

The execute phase of a compile-and-go operation is assigned the charge number associated with the compile. A compile-and-go operations occurs if the LIBRARY or SYNTAX option is not used with the COMPILE (Compile Program) command. An object code file will be generated and executed, and then removed from the system unless SAVE is specified.

CHARGE associates a program with a charge number but does not affect the user's or the program's security attributes.

Examples

EXECUTE A; CHARGE 670345

executes a program called A and inserts a user charge number of 670345 for it in the RLOG.

PERFORM LIST A; CG 5

invokes the DMPALL utility to print the contents of a file called A and inserts a user charge number of 000005 for it in the RLOG.

See also:

COMPILE (Compile Program) and USER (Assign Access Code) commands.

The USE CHRG record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The various language manuals for information about the ZIP facility.

CHECK (Verify Library Tape)

MCP Control Instruction

Access Level: 5

The CHECK command verifies the validity of a library tape.

It is recommended that you use the COPY command instead of the CHECK command, especially the COMPARE, CHECK, and SUMMARY options. The CHECK command will become unavailable at a future release.

The CHECK command starts a program called LOADMP, if the files are to be checked on disk, or a program called PACKUP, if files are to be checked on pack. LOADMP and PACKUP must be written by the user; Unisys does not supply them. If the appropriate program is not present on disk, the CHECK command is ignored.

See Appendix B for CHECK syntax, if you have user-coded LOADMP and/or PACKUP utilities.

See Also:

COPY (Invoke File Transfer Utility) command. Appendix B.

CK (Test Peripheral Device)

Keyboard Input Message

Access Level: 7

The CK command tests the status of a peripheral unit by using the TEST I/O operation. Any peripheral device except a reader-sorter (SOR, S4A, or S4B) or a non-status device (NST) can be checked with the CK command.

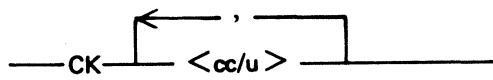


Figure 2-33. CK Command Syntax

<cc/u>

is the channel number and unit number of the device to be checked.

Details

The MCP issues a Test I/O to the device. The MCP displays four words (that is, four groups of four digits) of result descriptor (R/D) information. An R/D is a coded description of the results of an input/output operation.

Example

CK 6/3, 6/4

checks the status of the peripheral devices on channel 6, unit numbers 3 and 4.

See also:

The *V Series Programming Reference Manual* for information about read-sorter interfaces.

CN (Display Tape Number)

Keyboard Input Message

Access Level: 7

The CN command displays the five-digit physical tape number contained in a magnetic tape label.

— CN <cc/u> —————|

Figure 2-34. CN Command Syntax

<cc/u>

is the channel number and unit number of the tape unit. <cc/u> must reference a magnetic tape unit.

Details

You can use the AC (Place Tape Number In Label) or SN (Place Tape Number In Label) command to insert a physical tape number into a magnetic tape label.

Example

CN 2/1

displays the physical tape number in the label of the magnetic tape on channel number 2, unit number 1.

See also:

AC (Place Tape Number In Label) and SN (Place Tape Number In Label) commands.

COMPARE (Compare Files On Different Media)

MCP Control Instruction

Access Level: 5

COMPARE invokes the file transfer utility SYSTEM/COPY to compare the contents of files on different media. See the COPY (Invoke File Transfer Utility) command for details about syntax and how to use this command.

See Also:

COPY (Invoke File Transfer Utility) command.

COMPILE (Compile Program)

MCP Control Instruction

Access Level: 1

COMPILE executes an assembler or compiler to generate an object program from source language images.

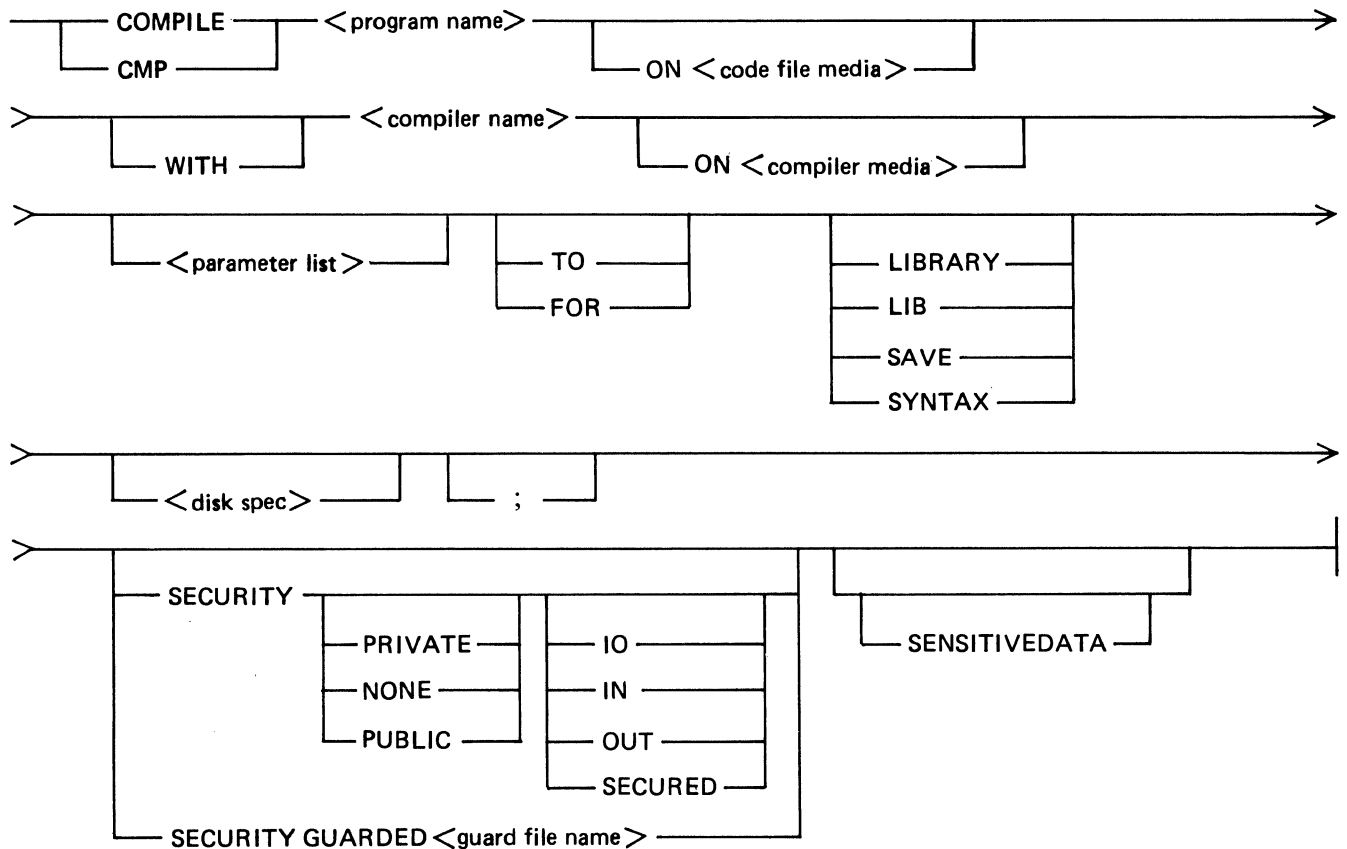


Figure 2-35. COMPILE Command Syntax

CMP

COMPILE

are synonymous.

<program name>

is the name of the code file to be created.

<code file media>

specifies where the code file will reside after it is created. The available options are as follows; if no option is specified, the default is to create the code file on 100-byte disk.

DISK

specifies creation of the code file on the default disk subsystem.

PACK

specifies creation of the code file on a system resource pack.

A pack family name
specifies creation of the code file on the given pack family name.

NOTE

The <code file media> can specify a pack only if the compiler used is capable of writing code files to pack. Some older compilers (such as COBOLV and BPL) lack this feature.

WITH

is optional and does not influence the command.

<compiler name>

is the name of the program that creates the code file.

<compiler media>

informs the MCP where to look for the compiler code file.

The CONTROL CODEPACK record that can be included at cold start in the system configuration file affects the sequence of attempts to find the compiler for execution. If ON <compiler media> is not included, the MCP will first look for the compiler on 100-byte disk. If not found, the MCP will look for the compiler on the pack specified in the CONTROL CODEPACK record, if specified. Otherwise, all non-restricted packs will be searched.

The allowable values for <compiler media> are as follows:

DISK

causes the MCP to look for the compiler code file only on disk.

PACK

causes the MCP to look for the compiler code file among all available non-restricted packs regardless of the presence of a CONTROL CODEPACK declaration at cold start.

A pack family name

causes only the specified pack family to be searched.

<parameter list>

are up to three parameters that can be inserted into a program when the program is started.

The <parameter list> option appears in the syntax of the COMPILE (Compile Program), DEBUG (Debug Session), and EXECUTE (Execute Library Program) commands. These parameters allow you to enter Boolean, string, and integer values, which can be used by the program when the program is being compiled or executed.

The syntax for <parameter list> is shown in Figure 2-36.

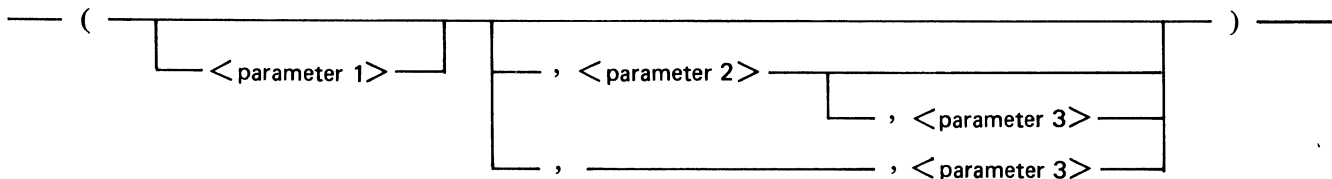


Figure 2-36. Syntax For <parameter list>

The rules for <parameter list> resemble those for parameters in the Work Flow Language (WFL) RUN command.

Integer parameters:

- contain unsigned numeric data up to eight digits in length (8 UN)
- have values ranging from 0-F
- are right-justified and zero-filled in the high-order digits if less than the maximum length
- will cause a syntax error if greater than the maximum length

String parameters:

- contain alphanumeric data up to six characters in length (6 UA)
- must be within quotes. To pass a quote as a parameter, a double quote must appear in the parameter string (for example, COMPILE MYPROG COBOL9 („TRY""XX").
- may be composed of any EBCDIC characters
- are left-justified and blank-filled in the low-order bytes if less than the maximum length
- will be truncated in the low-order bytes if greater than the maximum length

<parameter 1>

can be an integer up to eight digits in length. It sets the switches at program address 0. For example, CMP MYPROG COBOL9 (10100000).

<parameter 2>

<parameter 3>

can be eight-digit integers or six-character strings. <parameter 2> is inserted into the code file at base-relative address 8; <parameter 3> at base-relative address 20. For example, COMPILE MYPROG WITH COBOL9 (,12345678) or COMPILE MYPROG WITH COBOL9 (101,„TOTEST").

When either <parameter 2> or <parameter 3> are specified, you must have either a value or at least a comma in the preceding parameters. For example, COMPILE MYPROG WITH COBOL9 („,„TAPE").

A null entry is allowed. For example, COMPILE MYPROG WITH COBOL9 (,0,„ABC"" ") does not change the switches and inserts 00000000 at address 8 and the string "ABC" " at address 20.

TO
FOR

are optional and do not influence the command.

LIBRARY
LIB

specify that the new code file be made a permanent file on the specified disk or pack after an error-free compile. The code file is not active until explicitly executed.

SAVE

specifies that at the completion of an error-free compile the new code file is to be made a permanent file on the specified disk or pack. In addition, it is automatically executed.

SYNTAX

specifies that this is a compile for syntax-checking only. If no errors exist, the code file is neither saved nor executed.

If you do not specify LIB, SAVE, or SYNTAX, the program will be compiled and executed. (This is referred to as compile-and-go.) If syntax free, the object file will be removed at EOJ.

<disk spec>

is a one- or two-digit integer specifying the logical ID number on which the new code file is to reside (values 01-39) or the disk subsystem on which it is to reside (values 90-98). Note that <disk spec> can be included only if the code file is to be created on a 100-byte medium.

SECURITY

indicates that security attributes are going to be applied to this file.

PRIVATE

means that the file can be accessed only by the creator of the file in the manner specified by the SECURITYUSE attribute. PRIVATE is the default SECURITYTYPE attribute.

NONE

specifies that no security attributes are to be applied to this file.

PUBLIC

means that the file can be accessed by all users in the manner specified by the SECURITYUSE attribute.

IO

means that the SECURITYUSE attribute will allow READ/WRITE access to the file. IO is the default value assigned to SECURITYUSE.

IN

specifies that only READ access is allowed to the file.

OUT

specifies that only WRITE access is allowed to the file.

SECURED

means that the file is a program and can only be executed. Data cannot be read from or written to this file.

SECURITY GUARDED <guard file name>

specifies that another file identified by <guard file name> must be accessed before a user can be given permission to use the file. The SECURITYUSE attribute for a guarded file is stored in the GUARD file.

SENSITIVEDATA

specifies that all records of the file are to be overwritten with random data before the file is returned to the available disk pool. SENSITIVEDATA will be ignored if the file is declared as PUBLIC.

Slash Parameters

The <parameter list> option replaces former MCP slash parameters. MCP/VS 2.0 continues to allow slash parameters. However, when they are used, a warning will be displayed on the ODT and written to the ODT log stating that slash parameters will be deimplemented at the next MCP/VS release.

If you want to use slash parameters instead of <parameter list>, use the syntax shown in Figure 2-37.

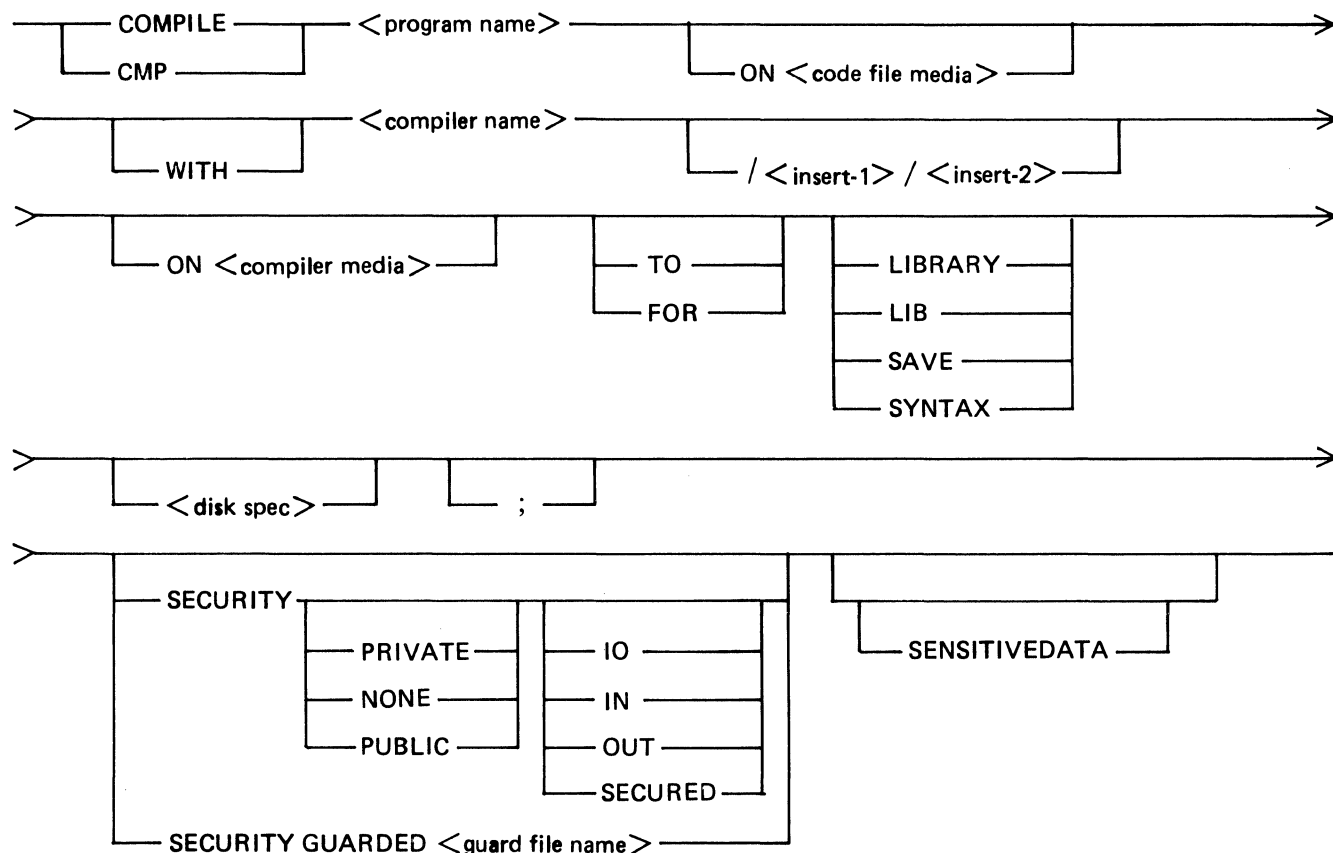


Figure 2-37. COMPILE Command With Slash Parameters Syntax

/<insert-1>/<insert-2>

are 1- to 6-byte alphanumeric parameters that can be used for various compilers.

The first parameter, padded with trailing spaces to total six bytes, is placed into the program at base-relative address 8.

The second parameter, if present and padded to six bytes, is inserted at base-relative address 20.

The area where these parameters are inserted is the same area as the index registers. Therefore, the parameters must be interrogated or saved before using the index registers for subscripting, indexing, or procedure entry, or any other purposes.

Getting Programs In the Correct Format

You can use the COMPILE (Compile Program) command to create a program code file on pack in the correct 180-byte code file format.

The PCOPY utility, which is described in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*, can be used copy a 100-byte disk format program code file to pack in the correct 180-byte code file format.

Examples

COMPILE OBJPGM ON SYSPAK WITH COBOL9 ON SYSPAK TO LIBRARY

compiles OBJPGM, which resides on a pack called SYSPAK, with the COBOL9 compiler, which is also on SYSPAK. If the compile is error-free, the new code file, OBJPGM, will become a permanent file on SYSPAK.

CMP TEST RPG SYNTAX

compiles TEST with the RPG compiler only to check syntax. The code file is neither saved nor executed.

CMP OBJ001 COBOL9 LIB ; SECURITY PRIVATE SENSITIVEDATA

compiles OBJ001 with the COBOL9 compiler. If it is an error-free compile, the new code file, OBJ001, will become a permanent file. This file can be accessed only by the creator, and all records of the file will be overwritten with random data before it is returned to the available disk pool when it is removed.

See also:

EXECUTE (Execute Library Program) command.

The CONTROL CODEPACK records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The PCOPY utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

COPY (Invoke File Transfer Utility)

COMPARE (Compare Files on Different Media)

COPYADD (Transfer Files)

MERGE (Transfer Files from Multiple Sources, Resolving Duplicates)

MOVE (Move Files to New Media)

MCP Control Instructions

Access Level: 5

These five commands invoke the SYSTEM/COPY utility, which transfers files and performs library maintenance functions between disk, pack, and tape.

See *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities* for more information about the SYSTEM/COPY utility.

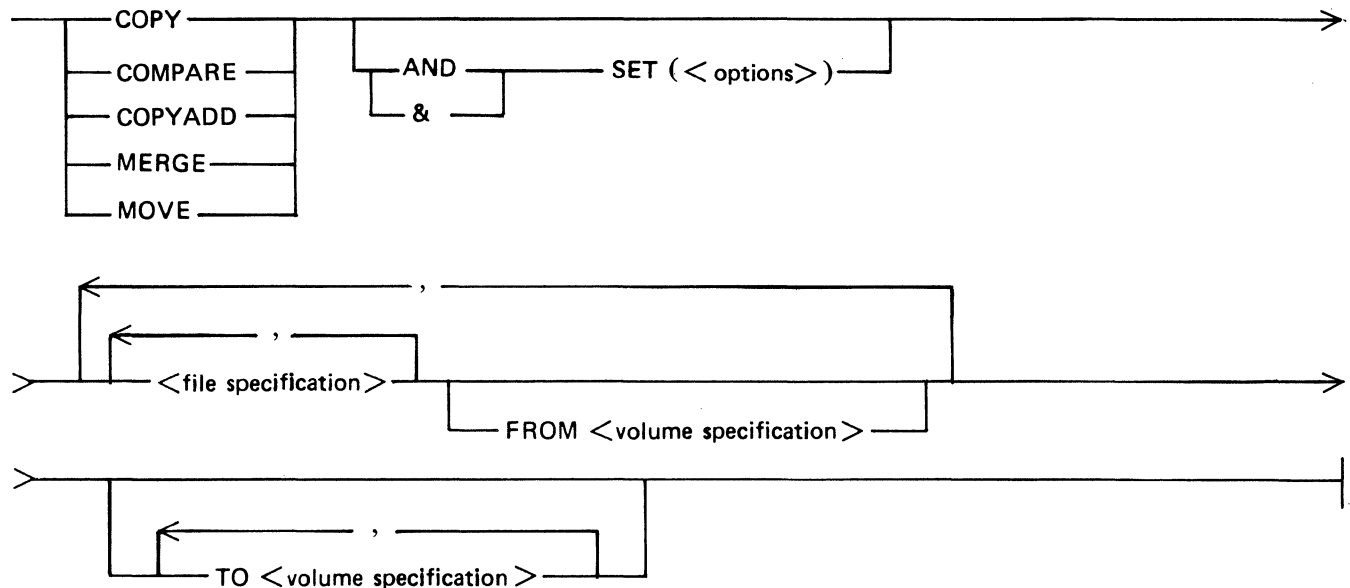


Figure 2-38. COPY, COMPARE, COPYADD, MERGE and MOVE Commands Syntax

COPY

copies the indicated files. If the files already exist at the destination, they will be replaced with the newly copied files.

Note that you cannot selectively copy files from multiple sources to multiple destinations.

COPYADD

copies only files that do not already exist at the destination. This is not a valid function if tapes are being used as destination media.

If AS is used to rename files as they are copied, the new names are used in the check for duplicates.

COMPARE

is both a keyword and an option. As a keyword it is used primarily when a large number of files are being copied to tape.

Both the option and the keyword compare *newly* copied files to their original source files and issue error messages in case of discrepancies.

The keyword can be used to compare files on one reel as more files are being written to another reel. For example, you can compare the files that were just written to reel number three while writing more files to reel number four.

A comparison of files on tape can start as soon as SYSTEM/COPY has finished writing files to that reel. If the COMPARE option is used, the comparison will not start until all the files have been copied to all the reels.

COMPARE as a keyword requires source and destination tapes to have identical directories (that is, to have the same files in the same order). Its use is based on the assumption that the two tapes were just copied.

MERGE

copies files from any number of sources to up to 10 destinations, resolving any duplicates. If the same file exists on more than one source, it is only copied from the last source specified in the command string on which it exists.

If the same file exists more than once on a given source tape, the SPECIFY option must be used to ensure that duplicates do not occur on any tape destination.

If AS is used to rename files as they are merged, the new names are used in the check for duplicates.

MOVE

copies files and then removes them from their sources. Files will not be removed from tape.

AND

&

are synonymous.

SET

causes the options listed to be set.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

<options>

can be used to complement the keywords. More than one option can be used at a time, as long as options are separated by a comma. Each option or string of options must be enclosed in parentheses. Underlining indicates the minimum abbreviation for each option. The syntax for these options is shown in Figure 2-39.

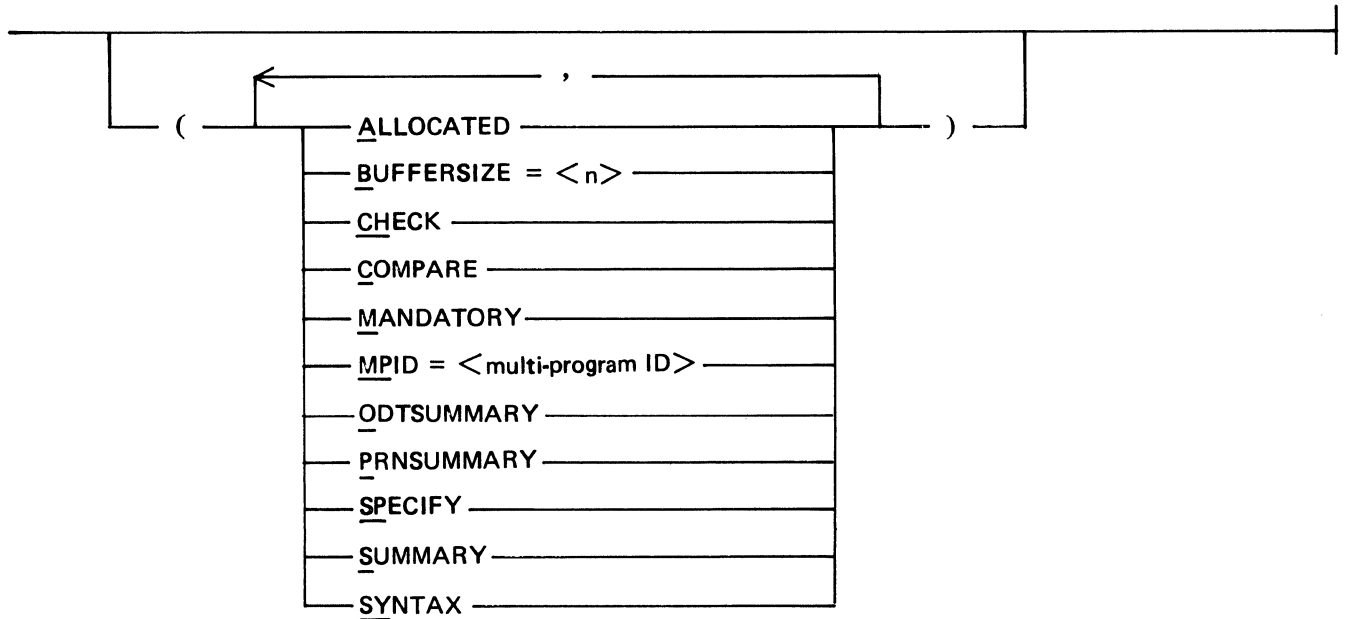


Figure 2-39. Command Options Syntax.

NOTE

All options except MPID and BUFFERSIZE can be Boolean expressions. Therefore, SET (COMPARE) is equivalent to SET (COMPARE=TRUE). The following diagram illustrates how to declare Boolean options.

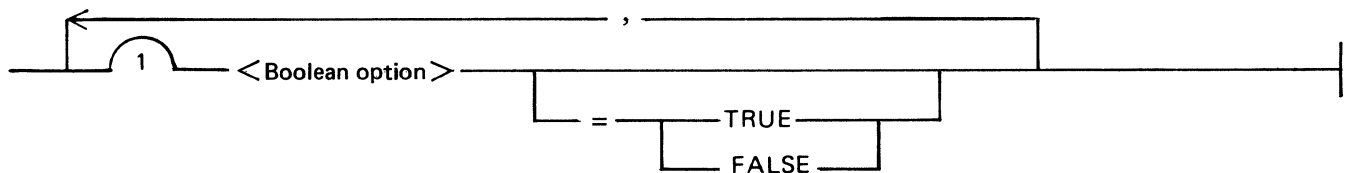


Figure 2-40. Boolean Options.

ALLOCATED

copies all source files onto predesignated sectors of disk or pack. All source files must already be present on all destinations. The source and destination files must have the same blocksize, blocks per area, and number of areas declared. End of File can be different.

Use of this option results in the destination file being opened input/output, and the source file then being copied onto the sectors already occupied or "allocated" by the destination file. Any additional sectors needed will be acquired, and any unused sectors will be returned.

This option does not allow a tape as a destination. The source and destination cannot be identical.

BUFFERSIZE

allows the user to specify the size of the internal buffers used by SYSTEM/COPY. When copying to tape, this sets the blocksize that is written to tape. The acceptable range in digits is $3600 < n < 99000$. The default value is 18000. SYSTEM/COPY will force to mod 1800 any value that is input.

CHECK

checks newly copied files for parity errors. If the file is on tape, COPY also checks for block count errors. If the file is on disk or pack, COPY also checks for end of file errors.

COMPARE

is both a keyword and an option. It compares data between source and destination files and sends a message to the ODT if there are discrepancies. It is highly recommended that you use this option with the keyword MOVE because files will not be removed from their sources if there are discrepancies between the original and the new files.

If an operation contains both a COMPARE option and an AS specification, the COPY function compares the original file to the new, renamed file.

MANDATORY

causes SYSTEM/COPY to require operator intervention if files are missing. If this option is not set, notification of missing files is still given, but the execution will continue.

MPID = <multi-program id>

gives different executions of COPY different names. For example, three COPY functions running simultaneously can appear on the ODT as LARRY/COPY, MOE/COPY and CURLY/COPY. Multiprogram identifiers can be up to six alphanumeric characters.

ODTSUMMARY

displays a summary of the operation on the ODT. The summary details the execution and the names of the files involved.

PRNSUMMARY

outputs a summary of the operation to a print file. The summary details the execution, error messages, operator responses, and the serial numbers of any tapes involved in the operation.

SPECIFY

allows the user to resolve multiple occurrences of the same file on a given source tape. This option is valid only with the keyword MERGE. For each file in a tape directory, a check is made for any duplicates against the remaining files in that directory.

If a duplicate is found, the following message will be displayed, and the user is given the opportunity to specify which file is desired.

```
SYSTEM/COPY=<mix number> TWO OCCURRENCES OF FILE <file name>  
                           REQUESTED FROM <tape name> (TAPE)  
                           AX "E" TO SPECIFY EARLIER OCCURRENCE  
                           AX "L" TO SPECIFY LATTER OCCURRENCE
```

**** SYSTEM/COPY=MX ACCEPT**

After a choice is made, the check will then continue looking through the rest of the directory. If another occurrence is found, the message is again displayed, and the choice of EARLIER or LATTER is given. EARLIER now refers to the previously chosen file. This process will continue for each additional duplicate occurrence of the file found.

SUMMARY

displays a summary of the operation on the ODT and also generates a print file that details the execution, error messages, and operator responses.

SYNTAX

causes SYSTEM/COPY to parse your commands without executing them. If there are any errors in the input, error messages will be displayed. If there are no errors, SYSTEM/COPY goes to EOJ.

<file specification>

indicates the files involved in an operation, and also changes the names of files as they are being copied. If more than one file is involved in an operation, their names must be separated by commas. The syntax for the file specification clause is shown in Figure 2-41.

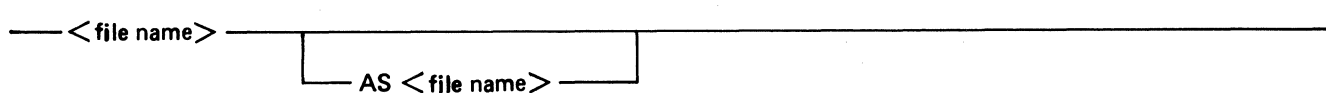


Figure 2-41. File Specification Syntax.

Group file names:

File names can be unique names (MYFILE), masked names (TEST=), or total group names (=).

Note a subtle difference when using total group names for selection of files from disk or pack media. The specification of a single equal sign ... = ... excludes all special system files (control deck, backup, pass, and dump files). On the other hand, three equal signs ... === ... includes them.

CAUTION

Unless MERGE is used, no checking is done to prevent duplicate files. COPY = AS A= FROM DISK TO PACK will replace the first character of each destination file name with an "A". In this case, both source files X123 and Y123 would be renamed A123, and therefore one will overwrite the other.

Tape File names:

Tape destination files can have multilevel file names through the use of the AS option.

The maximum size for multilevel file names is 10 levels of 17 characters per level. Slashes ... / / ... are reserved for level separation. Quotes ... " " ... and parentheses ... () ... are used for enclosing ID's with special characters. Equal signs ... = ... are used to mask file names and are only valid at the lowest ID level.

Tape files with multilevel IDs must be renamed using AS when they are reloaded from tape to disk or pack.

<volume specification>

indicates where to copy files FROM and where to copy files TO. Files from one or more sources can be copied to up to ten destinations. Copies of all source files are sent to all destinations. The syntax for the volume specification clause is shown in Figure 2-42.

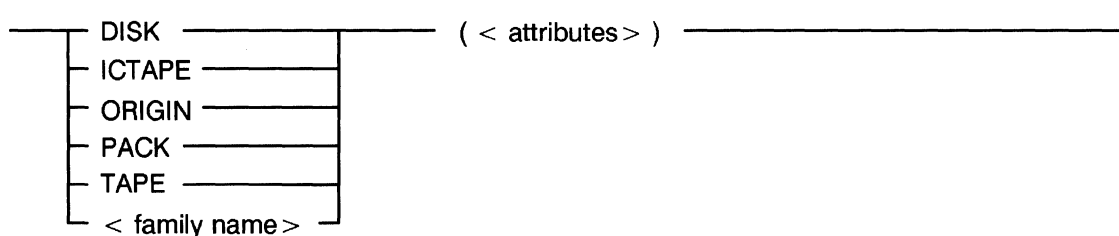


Figure 2-42. Volume Specification Syntax

DISK

is a generic family name for system resource disk.

PACK

is a generic family name for system resource pack.

TAPE

is shorthand for a general resource tape whose family name is SYSTEM. If a tape name is not specified when copying TO a tape, COPY will give it the name SYSTEM by default. If a name is not specified when copying FROM a tape, COPY will look to copy from a tape named SYSTEM.

ICTAPE

is shorthand for a general source tape whose family name is SYSTEM. If the name for an ICTAPE is not specified when copying TO an ICTAPE, COPY will give it the name SYSTEM by default. If the name of an ICTAPE is not specified when copying FROM an ICTAPE, COPY will look to copy from a tape named SYSTEM.

TAPE Versus ICTAPE

Two tape formats exist, identified by the volume specifications TAPE and ICTAPE. The TAPE format requires less tape than ICTAPE and is read and written faster. Only ICTAPE, however, is compatible with older releases of SYSTEM/COPY (those prior to the 6.7.1 releases of MCPIX and MCPVI).

Tape types are distinguished automatically on input.

ORIGIN

specifies the return of files from a tape to their origin. In other words, files copied from disk will return to disk, and those files copied from pack will return to pack. This is valid only for destination volume specifications.

<family name>

specifies a particular pack or tape.

<attributes>

further specify source and destination media. The syntax for specifying attributes is shown in Figure 2-43.

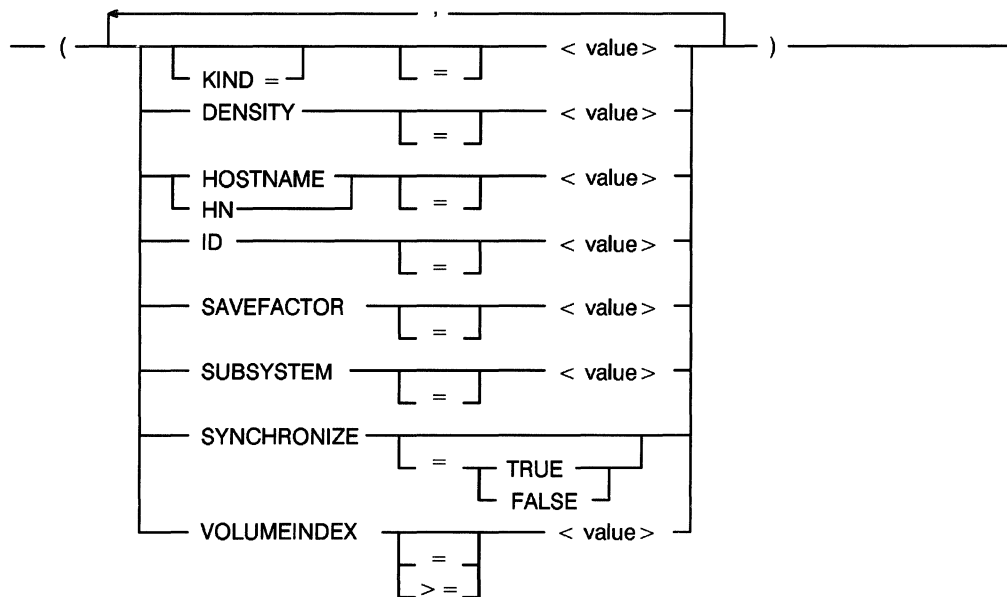


Figure 2-43. Attributes Syntax

The supported attributes and their values are:

■	KIND	DISK, PACK, TAPE, ICTAPE, TAPE9, TAPEPE, TAPEGCR, TAPEMTC, OLDTAPE
■	DENSITY	BPI800, BPI1600, BPI6250, BPI38000
	HOSTNAME, HN	1 to 17 character identifier
	ID	1 to 39
	SAVEFACTOR	1 to 999
	SUBSYSTEM	0 to 8
■	SYNCHRONIZE	TRUE, FALSE
	VOLUMEINDEX	1 to 999

KIND

specifies hardware types and is required for all user-named volumes. The volume names DISK, PACK, and TAPE have implied hardware types and the KIND attribute is not required to use them. Their values are:

■	DISK	(100-byte media)
	PACK	(180-byte media)
■	TAPE/ICTAPE	(9 track NRZ, PE, GCR, or MTC)
	TAPE9	(9 track NRZ only)
	TAPEPE	(PE only)
	TAPEGCR	(GCR only)
■	TAPEMTC	(MTC only)
	OLDTAPE	(a LOADMP or PACKUP tape)

SYSTEM/COPY does not support 7-track tape.

The **KIND =** syntax can be omitted. For example, (**KIND=PACK**) is equivalent to (**PACK**).

A specified value will override any implied value in the case of **DISK** and **PACK**. In other words, **DISK (KIND=PACK)** will be interpreted to mean a pack whose family name is **DISK**.

SYSTEM/COPY automatically recognizes and reads **LOADMP**, **PACKUP**, **TAPE**, and **ICTAPE** formats, so it is not necessary to indicate what kind of tape **SYSTEM/COPY** will be copying from. A volume kind of (**TAPE**) is all that needs to be specified when copying from tape.

While **SYSTEM/COPY** automatically recognizes and reads those tape formats, it writes only in the **TAPE** and **ICTAPE** formats.

TAPE9, **TAPEPE**, or **TAPEGCR** will result in a **TAPE** format tape of the corresponding density. To create an **ICTAPE** with a specific density, you must use the **DENSITY** attribute described below, as well as **KIND = ICTAPE**. For example: **COPY FILEA TO MYTAPE (ICTAPE, DENSITY = BPI6250)** results in a tape of **ICTAPE** format and **GCR** density.

DENSITY

allows you to specify density when **KIND = TAPE** or **ICTAPE** are indicated. The allowed values are:

BPI800	(800 bits per inch - 9 track NRZ)
BPI1600	(1600 bits per inch - PE)
BPI6250	(6250 bits per inch - GCR)
BPI38000	(38000 bits per inch - MTC)

HOSTNAME

HN

are identical. They specify the **BNA** host where this volume is resident. They are valid for both source and destination volumes; the **HOSTNAME** can be up to 17 characters long.

ID

specifies the logical disk unit. This attribute is restricted to disk destinations and has the range 1 to 39. All file areas will be assigned to the specified **ID**, with missing **ID** and no user disk conditions being resolved by the **MCP**.

SAVEFACTOR

specifies the number of days beyond the creation date that a destination tape can be automatically purged by the **MCP**. It ranges from 1 to 999. The default value is 999.

SUBSYSTEM

specifies the disk subsystem to be used for this volume. This attribute is restricted to disk destinations and has the range of 0 to 8 (0 being the default subsystems). All file areas will be assigned to identifiers defined within the specified subsystem, with undefined subsystem and no user disk situations being administered by the **MCP**. If this option is not used, disk native files will be returned to their original subsystems if possible.

■ SYNCHRONIZE

determines whether data is transferred between the host and tape drive in buffered or unbuffered mode.

- When SYNCHRONIZE is FALSE, transmission is buffered (asynchronous). This maximizes tape drive performance.
- When SYNCHRONIZE is TRUE, transmission is *unbuffered*.

This attribute applies only to magnetic cartridge tape devices (KIND = TAPEMTC) and tape drives that use the Small Computer Systems Interface (SCSI).

When SYNCHRONIZE is not entered in a volume specification, the default value is FALSE. To set SYNCHRONIZE to TRUE, enter SYNCHRONIZE by itself or SYNCHRONIZE = TRUE.

VOLUMEINDEX

applies to source tape volumes and is used to isolate a single reel of a volume set. Because each reel contains a directory, a file required from the third reel of a volume set can be loaded directly from that reel instead of having to sequentially process the first two reels. This attribute can also be used to overlap transfer of a complete volume set by initiating separate SYSTEM/COPY executions, each loading an exclusive VOLUMEINDEX. This attribute has the range of 1 to 999.

VOLUMEINDEX >= <value> can be used to load files from reel <value> through the end of that volume set. For example, to load all the remaining files from a volume set starting at reel number 2, use the syntax VOLUMEINDEX >= 2.

Examples

COPY FILE1 AS FILE2

copies a file named FILE1 from system disk to system disk, renaming the file FILE2. Because both the source and destination volumes were not named, the system uses the system disk by default.

COPY AND SET (ODTSUMMARY, MPID=JOHN) MYFILE TO DEPACK (PACK)

copies a file named MYFILE from disk to a pack named DEPACK, and sends a SUMMARY of this operation to the ODT. The MPID option gives this execution of SYSTEM/COPY the unique name JOHN/COPY.

MERGE AND SET (SUMMARY, MPID=ANZA) = FROM TAPE1 (TAPE), = FROM TAPE2 (TAPE) TO TAPE3 (TAPE)

places a copy of all the files from TAPE1 and TAPE2 on TAPE3. If TAPE1 and TAPE2 have duplicate files, the duplication is resolved by taking the file from TAPE2, the latter tape. An ODT and printer summary are produced and the program will be in the mix as ANZA/COPY.

COPY X= TO TAPEA (ICTAPE)

copies all files whose names begin with "X" from system disk to a tape, names the tape "TAPEA", and writes the tape in ICTAPE format.

COPY = FROM TAPEA (TAPE)

copies all the files on the tape named TAPEA to system disk. The COPY command automatically recognizes tape formats, making it unnecessary to specify the type of tape.

COPY AND SET (COMPARE) FILE1, FILE2 FROM FINBAR (PACK)

copies two files named "FILE1" and "FILE2" from a pack named "FINBAR" to system disk. The system uses the system disk by default because no destination media is defined. After all the copying is complete, the system compares the new files to their original files, and sends an error message if there are any discrepancies between the new and original files.

COPY AND SET (SUMMARY) = FROM SYSTEM (TAPE), = FROM UPDATE (TAPE) TO DISK

copies all files from a tape named "SYSTEM" and a tape named "UPDATE" to system disk. In addition, the system sends a summary of the operation to the ODT and another, more detailed summary to a print file.

If any of the tape files already exist on disk, they will be replaced with the new files. If both tapes contain a file with the same name, only the file from the last tape named as a source will be copied.

MOVE AND SET (COMPARE) A=, B= FROM DISK, C= FROM PACK, D= FROM KTEL (PACK) TO MORGUE (PACK)

transfers all the files whose names begin with A and all the files whose names begin with B from system disk to a pack named MORGUE; also transfers all files whose names begin with C from any unrestricted pack(s) to a pack named MORGUE, and transfers all files whose names begin with D from pack KTEL to a pack named MORGUE.

After all files have been transferred, the compare option causes each newly copied file to be compared to its source file for accuracy.

COPY A= FROM MORGUE (PACK), B= FROM DISK TO MYTAPE (ICTAPE)

transfers all files whose names begin with A from a pack named MORGUE to MYTAPE, and transfers all files whose names begin with B from the system disk to MYTAPE. This command also instructs the system to write MYTAPE in ICTAPE format.

COPY = FROM MYTAPE (TAPE) TO ORIGIN.

transfers all the files on a tape named MYTAPE back to the particular media from which they came. (The function invoked by ORIGIN works only when files are being transferred FROM tape.)

See also:

The discussion of SYSTEM/COPY in the V Series MCP/VS System Software Operations Guide Volume 3: System Utilities.



COPYADD (Transfer Files)

MCP Control Instruction

Access Level: 5

COPYADD invokes the SYSTEM/COPY utility to transfer files from one location to another. Only files whose names are not already present on the destination medium are transferred.

COPYADD cannot specify tape as an output medium.

See COPY (Invoke File Transfer Utility) for details about syntax and how to use the command.

See also:

COPY (Invoke File Transfer Utility) command.

/OWN

Using the /OWN option prevents the scheduling of the bound MCP intrinsic when a user-provided LDCNTL is required but is not on disk.

If this option is omitted, the user-coded version is initiated if present on disk, otherwise the MCP intrinsic is initiated.

informs LDCNTL that further specifications, such as skipping lines or restarting, will be provided through an AX response after the LDCNTL program begins operation.

After locating the specified file, LDCNTL displays "ENTER SPECS" and issues an ACCEPT. The response to the ACCEPT must be in one of the formats shown in Figures 2-167, 2-168, or 2-169 under the PRINT (Print Files) command.

=

indicates that all punch backup files on the specified media will be converted.

<copies>

indicates the number of copies of the file you want. One copy is the default.

Details

The LDCNTL utility is a bound intrinsic or a user-coded program that creates pseudo decks from physical card decks. Pseudo card decks are created with the LD (Create Pseudo Card Deck) or EXECUTE LDCNTL commands. Use the CV or CVP command only if you want to convert a punch backup file into a pseudo card deck.

Examples

CV /MCP *10023 SAVE

converts punch backup disk file *10023 into a pseudo card input deck and causes the file not to be removed after the conversion.

CVP /OWN =; CG 67073

starts a user-coded version of the LDCNTL utility to convert all punch backup pack files into pseudo card decks. A user charge number of 67073 for this program into the MCP Run log records.

See also:

EXECUTE (Execute Library Program); LD (Create Pseudo Card Deck); PC, PCP (Create Punch Backup File); PRINT (Invoke PBDPRN); and SD (Set Deck Limit) commands.

The LDCNTL utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities* and the *V Series MCP/VS Programming Reference Manual*.

DA, DAP (Deactivate Pseudo Reader)

Keyboard Input Message

Access Level: 5

The DA command deactivates a pseudo card reader on disk or pack.

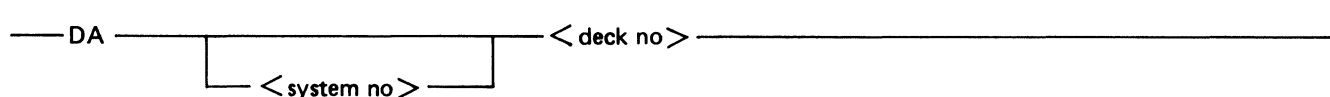


Figure 2-46. DA Command Syntax (Format 1 - Disk)

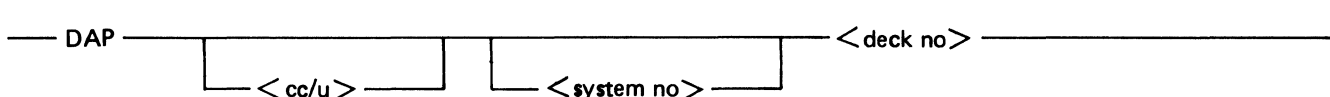


Figure 2-47. DA Command Syntax (Format 2 - Pack)

Format 1

DA

is used to deactivate pseudo card readers on disk.

<system no>

If this is omitted, the number of the originating system is used.

<deck no>

must reference a previously activated pseudo card reader.

Pseudo decks are designated as #snnnn. The s represents the system number and is not required unless the deck was created by a system other than the one from which you are entering the command. The nnnn is a four-digit number with leading zeros assumed.

Do not include the pound sign in the command syntax.

You can determine activated pseudo card readers with a OL PCR command.

Pseudo card readers are activated either with the RN (Activate Pseudo Reader) command or automatically by the MCP if the APCR option has been set ON.

Format 2

DAP

is used to deactivate pseudo card readers on pack.

<cc/u>

indicates the channel number and unit number of the pack on which to deactivate pseudo card readers. If <cc/u> is not specified, the first deck encountered with the specified system number, if indicated, and deck number is deactivated.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

DA 153

deactivates pseudo card reader number 153 on disk.

DA 3 1

deactivates pseudo card reader on system number 3 with deck number 1.

DAP 6/0 1 32

deactivates pseudo card reader on the pack on channel number 6, unit number 0, from system number 1, deck number 32.

See also:

CD, CDP (List Inactive Pseudo Decks on Disk or Pack); OL (Display Peripheral Status); RD, RDP (Remove Pseudo Decks); and RN (Activate Pseudo Reader) commands.

The USE APCR record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

DATA/DATAB (Mark Beginning Of Card Data File)

MCP Control Instruction

Access Level: 0

The DATA/DATAB command indicates the beginning of a punched card data file.

Whenever the DATA/DATAB command is encountered in the card reader, it causes an End-Of-File (EOF) condition on any previous file, and the unit is marked by the MCP as containing the new file.

This command is valid only from an actual or pseudo card deck.

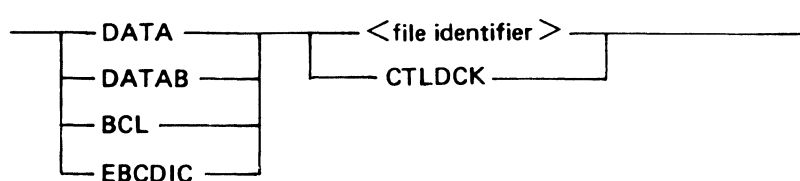


Figure 2-48. DATA/DATAB Command Syntax

DATA EBCDIC

signify that subsequent input is in EBCDIC format. EBCDIC stands for Extended Binary Coded Decimal Interchange Code, a coded character set in which each character is represented by an eight-bit code.

DATAB BCL

indicate that the file is in BCL format. The BCL language is a coded character set in which each character is represented by a six-bit code.

<file identifier>

indicates the name of the file to follow.

Any information following the file identifier on the same punched card is ignored.

CTLDCCK

is a reserved file identifier that allows valid Control records to be included in the file. CTLDCCK is the name of the physical card deck used by the LDCNTL utility to build a pseudo card deck.

NOTE

If <file identifier> is CTLDCCK, no EOF action is taken until the ENDCTL command is encountered. This is primarily intended for use by the utility program LDCNTL for building pseudo card decks.

Examples

DATAB INPUT3

specifies the beginning of a punched card data file called INPUT3. The file is in BCL format.

COMPILE PROG COBOL9 LIBRARY; DATA CARD

compiles the COBOL program PROG using the COBOL9 compiler and then saves the new code file as a permanent file on disk. Then, this statement specifies the beginning of a punched card data file called CARD. CARD is in EBCDIC format.

See also:

BCL (Mark Beginning Of Card Data File); BINARY (Read Binary Card Deck); EBCDIC (Mark Beginning Of Card Data File); and END/ENDCTL (Mark End of Card File or Control Instructions) commands.

DBCA (Clear SSP Address Contention)

Keyboard Input Message

Access Level: 8

The DBCA command directs the MCP to clear the system contention flag for the specified addresses in a Shared System Processor (SSP) on the system.

DBCA allows you to clear all SSPs and all address contentions for the specified system; to clear one SSP and all addresses for the specified system; or to clear one SSP and one address.

This command is not intended for use during normal operations; it has been implemented primarily for system debugging.

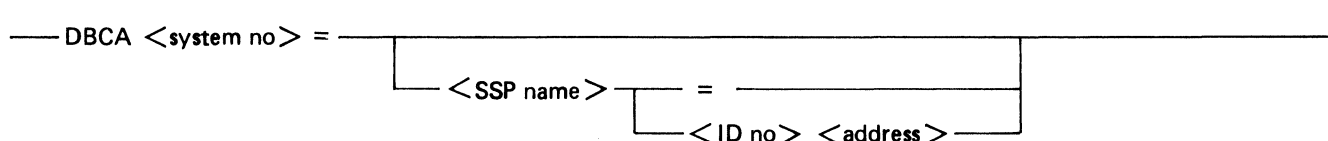


Figure 2-49. DBCA Command Syntax

<system no>

designates the system (0 through 3) whose contention must be reset.

DBCA <system no> =

clears all SSPs and all address contentions for the specified system.

<SSP name>

is the mnemonic name given to an SSP on the DLP SSP record of the system configuration file.

<SSP name> =

causes all address contentions for the specified system on the specified SSP to be cleared.

<ID no>

is the three-digit unit designation used to specify the ID number of the pack or disk device. The number can range from 1 to 255.

<address>

designates the specific address (range 0 to 16700000) to be cleared; the address is always decimal.

Details

An SSP is a type of Data Link Processor (DLP) that controls concurrent file access in a multi-system shared environment. The SSP stores addresses that have been locked in order to prevent simultaneous updating of shared files.

You can use the DBDR (Produce SSP Content Report) command or the SHOW SSP option of the SHOW (Display System Information) command to display on the ODT the current contents of all SSPs on the system.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

DBCA 1 SSPABC =

clears all address contentions in SSP SSPABC for system number 1.

DBCA 2 =

clears all address contentions in all SSPs on system number 2.

See also:

DBDR (Produce SSP Content Report) and SHOW (Display System Information) commands.

DBCL (Cancel DLP)

Keyboard Input Message

Access Level: 8

This command directs the MCP to fire an unconditional cancel at a specified Data Link Processor (DLP).

— DBCL <channel no> —

Figure 2-50. DBCL Command Syntax

<channel no>

represents the channel number of the DLP to be canceled.

Example

DBCL 6

fires a cancel at the DLP associated with channel number 6.

DBDR (Produce SSP Content Report)

Keyboard Input Message

Access Level: 8

The DBDR command provides a report on the ODT of the current contents of all the Shared System Processors (SSPs) on the system. This command is valid only if the system is equipped with an SSP.

DBDR is not intended for use during normal processing; it has been implemented primarily for system debugging.

— DBDR —|

Figure 2-51. DBDR Command Syntax

Details

The SHOW (Display System Options) command has a parameter that displays an SSP or all SSPs on the system.

If DBDR, if used, will be recorded in the ODT log. However, if SHOW SSP is used, the system responds to the command but does not log it.

See also:

DBCA (Clear SSP Address) and SHOW (Display System Information) commands.

DBIC (Initialize Controller Parameters)

Keyboard Input Message

Access Level: 8

DBIC enables, disables, or displays the status of a path from a B9387 controller to a dual exchange. It is valid only for sequential Data Link Processors (DLPs).

This command is used normally after analyzing the Maintenance log (MLOG) for the presence or absence of exchange paths retry result descriptors.

The DBIC status is remembered across halt/loads and after controller host loads.

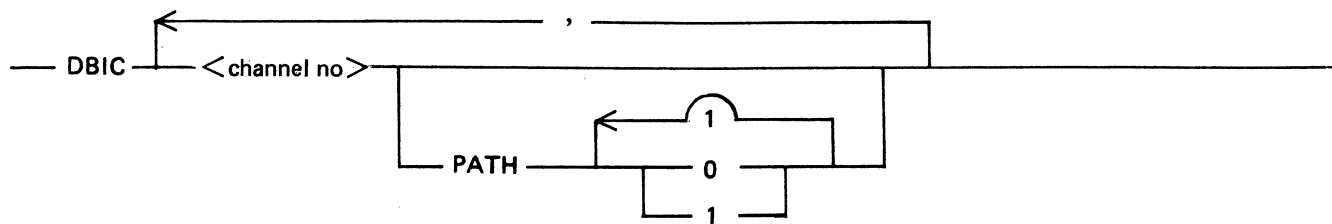


Figure 2-52. DBIC Command Syntax

<channel no>

is a valid channel number. If you do not specify a PATH option, the status of the specified channel will be displayed.

PATH

0

1

enables one or both paths. One path can be disabled by specifying the other path. For example, DBIC <channel> PATH 0 enables path 0 and disables path 1. Both paths are enabled by specifying both paths. For example, DBIC <channel> PATH 0 1 enables both paths.

Examples

DBIC 24, 7

displays the status of channels 24 and 7.

DBIC 24 PATH 0, 7 PATH 1

enables PATH 0 on channel 24 and PATH 1 on channel 7.

DBIC 24 PATH 0 1

enables PATH 0 and PATH 1 on channel 24.

See also:

The *V Series System Software Logging Operations Reference Manual* for information about the Maintenance log.

DBMM (Set/Reset Maintenance Mode)

Keyboard Input Message

Access Level: 8

DBMM sets or resets the maintenance mode option available on 206, 207, 659, and 677 type packs.

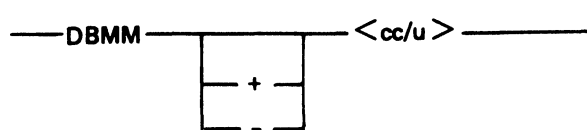


Figure 2-53. DBMM Command Syntax

- + sets maintenance mode, making the unit unavailable to the system.
- resets maintenance mode and can return the unit to system availability, depending on the unit's SAVED or RESERVED status.

<cc/u> is the channel number and unit number of the device. If you enter DBMM <cc/u>, the current maintenance mode status for the specified device is displayed.

Details

Putting a drive in maintenance mode makes it unavailable to the system and allows off-line testing of the drive.

Resetting maintenance mode makes the unit available for system operation. Maintenance mode can be reset only by issuing a DBMM - <cc/u> command or by powering off the drive locally.

The current maintenance mode setting is retained in the pack controller.

The DBMM status is remembered across halt/loads and is not affected by the UR (Inhibit or Disinhibit Unit) command, which inhibits units.

NOTE

This command is not valid for 680, MD4, or MD8 type packs or MCP disk.

Examples

DBMM + 6/3

sets maintenance mode on the pack on channel 6 unit number 3.

DBMM 4/2

displays the maintenance mode status of the pack on channel 4 unit number 2.

DBSR (Produce Block Lockout Table Report)

Keyboard Input Message

Access Level: 8

The DBSR command requests the MCP to provide a report of the current contents of the MCP Block Lockout Table (BLT).

The BLT is a table that keeps track of which file blocks are locked and/or contended for by which programs. The BLT resides in memory and is updated by the MCP every time a program locks, unlocks, or contends for a file block.

If the system is part of a multi-system shared configuration, the status of the SSP lock for each BLT entry is also given.

DBSR is not intended for use during normal processing; it has been implemented for system and program debugging.

— DBSR —

Figure 2-54. DBSR Command Syntax

Details

The SHOW (Display System Options) command has a parameter that displays an entry from a BLT or an entire BLT.

If DBSR, if used, will be recorded in the ODT log. However, if SHOW BLT is used, the system responds to the command but does not log it.

See also:

SHOW (Display System Options) command.

DBUA (Unlock Address)

Keyboard Input Message

Access Level: 8

DBUA initiates an unlock operation on the designated address. The DBUA command is valid only if the designated system is equipped with a Shared System Processor (SSP).

DBUA allows you to unlock all addresses and all SSPs for the specified system; to unlock all addresses on one SSP for the specified system; or to unlock one address on one SSP.

This command is not intended for use during normal operations; it has been implemented primarily for system debugging. It is useful for unlocking addresses held by a processor that has become unavailable for some reason.

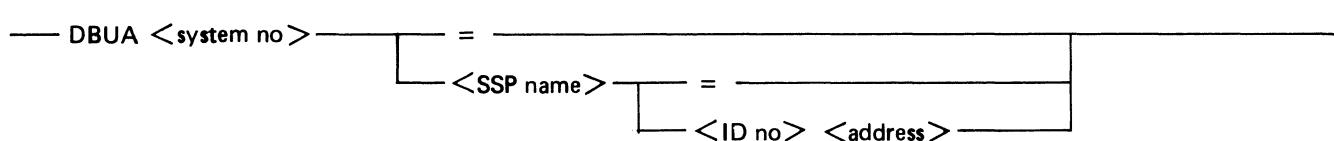


Figure 2-55. DBUA Command Syntax

<system no>
designates the system (0 through 3) for which you want to perform the unlock operation.

DBUA <system no> =
unlocks all SSPs and all addresses for the specified system.

<SSP name>
is the mnemonic name given to an SSP on the DLP SSP record in the system configuration file.

<SSP name> =
causes all addresses for the specified system on the specified SSP to be unlocked.

<ID no>
is the three-digit unit designation used to specify the ID number of the pack or disk device. The number can range from 1 to 255.

<address>
designate the specific address (range 0 to 16700000) to be unlocked; the address is always decimal.

Details

An SSP is a type of Data Link Processor (DLP) that controls concurrent file access in a multi-system shared configuration. The SSP stores addresses that have been locked in order to prevent simultaneous updating of shared files.

Examples

DBUA 0 =
unlocks all addresses and all SSPs for system 0.

DBUA 2 FRED 12 50000
unlocks address 50000 on ID number 12 in SSP FRED on system 2.

See also:

DBCA (Clear SSP Address Contention) command.

The DLP SSP record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

DBUL (Clear Locked Files on Shared Systems)

Keyboard Input Message

Access Level: 8

In a shared system environment a system can fail, leaving files locked on shared disk and pack media. The DBUL command directs the MCP to initiate a search for these files on disk and shared packs and clear the locked bits.

The DBUL command can be used if the following is true:

- the designated system allows concurrent file access
- the system is equipped with a Shared System Processor (SSP)

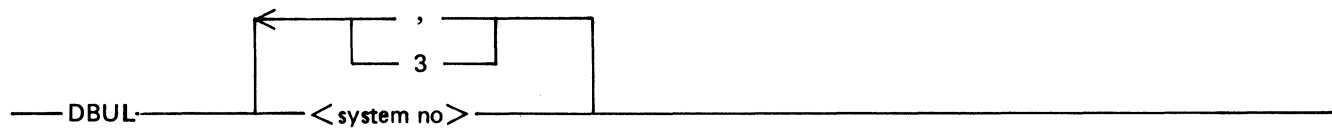


Figure 2-56. DBUL Command Syntax

<system no>

is the number (0 through 3) of the "down" system.

Examples

DBUL 0

clears locked files on system 0.

DBUL 1,2

clears locked files on systems 1 and 2.

See also:

DBCA (Clear SSP Address); DBSR (Produce Shared Disk Report); DBUA (Unlock Disk Address); and SHOW (Display System Information) commands.

The discussion of shared systems in *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

DC (Display Compilation Data)

Keyboard Input Message

Access Level: 1

The DC command displays on the ODT the program name, date compiled, compiler type (first character of the compiler name used to create this program), memory requirement, and the internal program name or compile time for a specified program residing on disk or pack.

DC can also be used for an Independently Compiled Module (ICM). ICMs are modules that can be bound together with other modules to create a single program. There are different types of ICMs. The types restrict the language(s) to which that ICM can be bound.

The display will provide the ICM name, version level, internal id (may contain compiler name and release number depending on ICM type), compile date, and time of compilation.

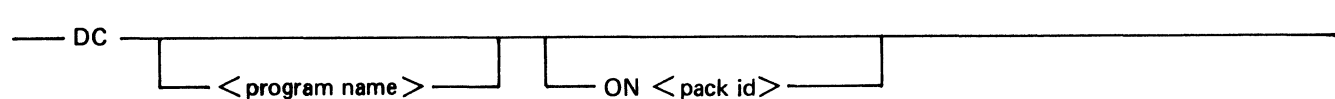


Figure 2-57. DC Command Syntax (Non-ICM Type)

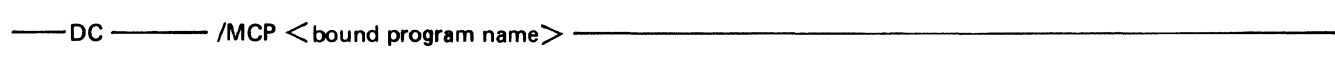


Figure 2-58. DC Command Syntax (Non-ICM Type)

DC

without any options causes the display to show the MCP name, release level, patch level, and date and time compiled of the currently running MCP.

<program name>

designates the name of the program whose compile information you want to display. A program name can have up to six characters. The first character must be a letter of the alphabet.

If you are using the /MCP option, specify a bound program name.

/MCP

indicates that the specified program is bound to the MCP.

ON <pack id>

indicates that the specified <program name> is located on a pack labeled with the name indicated by <pack id>.

If you are using the DC command to display a bound program, you cannot specify ON <pack id>. Bound programs are always on disk.

Details

The memory requirement specified does not include any additional space the program might require after it begins execution (for example, disk file header space).

Example

DC COBTY2

causes the ODT to display the program name, data compiled, compiler type, memory requirement and compile date/time.

NOTE

Some compilers display the time of compile rather than internal program name.

Displaying ICM Compilation Data

— DC <ICM name> —|

Figure 2-59. DC Command Syntax (ICM-Type).

<ICM name>

is the name of the ICM for which the DC is requested.

The ICM cannot be a FORTIV ICM, XFORTN ICM, or a BPL type 1 ICM.

Example

DC COBICM

causes the ODT to display the ICM name, version level, compiler name, internal-id, date and time of compilation.

See also:

DISPLAY/WHATS (Display Information), FN (List File Names), and WB (Display Bound Programs) commands.

DD (Delete DLP)

Keyboard Input Message

Access Level: 8

This command removes a DLP from the system environment. This allows you to change the firmware ID, hardware type, or other information about the DLP, and then re-declare the DLP.

The primary channel of an exchange is the first DLP declared on the EXCHANGE record; the primary channel has the same number as the exchange number. If the primary channel has no attached units, it can be deleted.

All other channels on an exchange are alternate channels. If the channel to be deleted is an alternate channel of an exchange, you must use the XA (Add/Remove a Channel) command first, and then the DD command to delete it.

DD <channel no>

Figure 2-60. DD Command Syntax

<channel no>

indicates the channel number of the DLP you are deleting. If the channel number is not present, the system displays:

DLP <channel number> NOT PRESENT

If you attempt to delete a primary channel that has units attached, the system displays:

UNIT xx/yy EXISTS ON THIS CHANNEL - CANNOT DELETE

You use the DL (Delete Peripheral Devices) command to delete units from the primary channel. You can then delete the primary channel.

See also:

DL (Delete Peripheral Devices) and XA (Add/Remove A Channel) commands.

DEBUG (Attach Program To DEBUG Session)

MCP Control Instruction

Access Level: 1

The DEBUG command starts a program and simultaneously attaches the ODT to a DEBUG session for that program if the DBUG option is set. This command is entered only through an ODT.

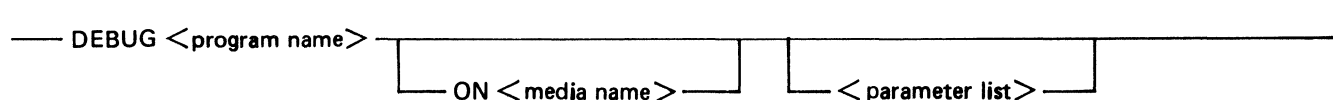


Figure 2-61. DEBUG Command Syntax

<program name>

is the name of the program that will be attached to an interactive debug session. The file must contain executable object code and must be stored on the disk or pack entered as <media name> if specified. Otherwise, it will be stored on disk.

<media name>

refers to the name of the storage device where the desired program is stored. The <media name> can be one of the following choices:

DISK

looks for the desired program on 100-byte media only. If the program is not found, an error message is displayed on the ODT.

PACK

looks for the desired program on all of the system's unrestricted packs. If the program is not found, an error message is displayed on the ODT. The program cannot be found unless it is stored in 180-byte code file format.

To ensure that the program is stored in the correct format, copy it onto the pack using the PCOPY utility or compile the program using the <code file media id> option of the COMPILE (Compile Program) command.

Pack Family Name

looks for the desired program on the packs of the entered family name. If the program is not found, an error message is displayed on the ODT.

If <media name> is not entered, the operating system will search all 100-byte disks for the desired program. If the program is not stored on disk, the operating system will continue searching for the program on all unrestricted packs. If the program is not found on disk or on unrestricted pack, an error message will be displayed on the ODT.

<parameter list>

are up to three parameters that can be inserted into a program when the program is started.

The **<parameter list>** option appears in the syntax of the **COMPILE** (Compile Program), **DEBUG**, and **EXECUTE** (Execute Library Program) commands. These parameters allow you to enter Boolean, string, and integer values, which can be used by the program when the program is being compiled or executed.

The syntax for **<parameter list>** is shown in Figure 2-62.

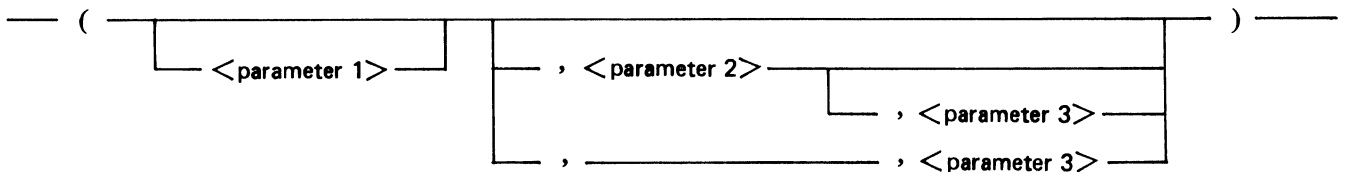


Figure 2-62. Syntax For <parameter list>

The rules for **<parameter list>** resemble those for parameters in the Work Flow Language (WFL) **RUN** command.

Integer parameters:

- contain unsigned numeric data up to eight digits in length (8 UN)
- have values ranging from 0-F
- are right-justified and zero-filled in the high-order digits if less than the maximum length
- will cause a syntax error if greater than the maximum length

String parameters:

- contain alphanumeric data up to six characters in length (6 UA)
- must be within quotes. To pass a quote as a parameter, a double quote must appear in the parameter string (for example, **DEBUG MYPROG ("TRY""XX")**).
- may be composed of any EBCDIC characters
- are left-justified and blank-filled in the low-order bytes if less than the maximum length
- will be truncated in the low-order bytes if greater than the maximum length

<parameter 1>

can be an integer up to eight digits in length. It sets switches at program address 0. For example, `DEBUG MYPROG (10100000)`.

<parameter 2>

<parameter 3>

can be eight-digit integers or six-character strings.

<parameter 2> is inserted into the code file at base-relative address 8; <parameter 3> at base-relative address 20. For example, `DEBUG MYPROG (,12345678)` or `DEBUG MYPROG (101,"TOTEST")`.

When either <parameter 2> or <parameter 3> are specified, you must have either a value or at least a comma in the preceding parameters. For example, `DEBUG MYPROG (,,"TAPE")`.

A null entry is allowed. For example, `DEBUG MYPROG (,0,"ABC"" ")` does not change the switches and inserts 00000000 at address 8 and the string "ABC"" " at address 20.

Details

The interactive `DEBUG` module is used to debug user programs or the operating system itself. `DEBUG` can:

- perform `DEBUG` sessions concurrently
- establish program breakpoints
- output `DEBUG` tracings to the ODT, a printer, tape, or to a printer backup file on disk
- debug all or part of a task
- allow modification of memory within the task being debugged

The `GT` (Initiate Trace) and `TRACE` (Trace A Program) commands start tracing facilities for user programs only. These tracing facilities do not have all the capabilities of the `DEBUG` module.

Related Commands

The `ID` command (Invoke `DEBUG` Session) starts a `DEBUG` session for a program that is already running and attaches the ODT to that `DEBUG` session.

The `ED` (Enter `DEBUG` Session) and `QD` (Query `DEBUG` Sessions) commands, like the `ID` command, manipulate `DEBUG` sessions. The `ED` command stops the ODT from displaying system information and returns it to a previously started `DEBUG` session. The `QD` command displays the number of active debug sessions on the system with their corresponding session numbers.

Effect On The Run Log

The `DEBUG` command sets the two-bit in the supplementary execution code field (RL-SXE, which is digit 71) of the job's Job Schedule record (type 3/0), the Short Job Schedule (type 4/0), and Beginning-of-Job record (record type 6/0) in the system Run log.

If one program starts another program using the `ZIP` construct, the `DEBUG` command can be included in the `ZIP` syntax. In this way an entire system of programs can be correctly flagged in the Run log, as well as the one originally executed.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

DEBUG PAYRL1

looks for a program named PAYRL1 on disk, CODEPACK, or unrestricted pack; starts the program; and attaches it to an interactive debugging session.

DEBUG UPDATE ON PACK

looks for a program named UPDATE on all unrestricted packs, starts the program and attaches it to an interactive debugging session.

DEBUG STAT01 ON ACCTS ; FILE OUTPUT = LINEPR/ST1985

looks for a program named STAT01 on the pack family named ACCTS, starts the program and directs the file named OUTPUT to specific printer.

See Also:

COMPILE (Compile Program); ED (Enter DEBUG Session); GT (Initiate Trace); ID (Invoke DEBUG Session); QD (Query DEBUG Sessions); SO (Set System Options); and TRACE (Initiate Trace) commands

The USE DBUG record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The DEBUG utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

The system Run log in the *V Series System Software Logging Operations Reference Manual*.

DISK (Add A Disk)

MCP Control Instruction

Access Level: 8

The DISK command adds a disk to the MCP environment. The syntax for the DISK command is identical to that for the DISK record of the system configuration file.

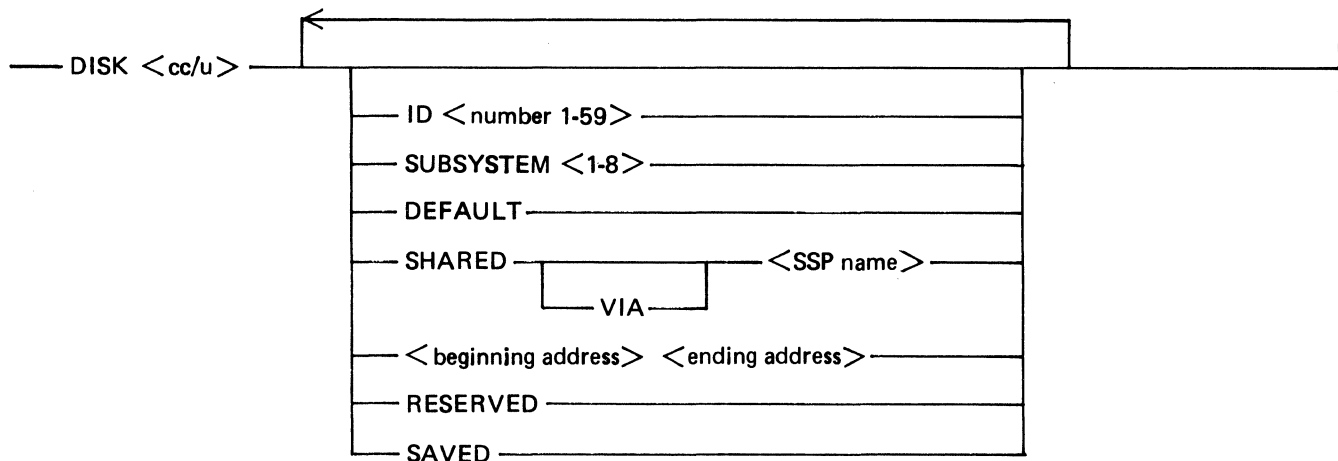


Figure 2-63. DISK Command Syntax

<cc/u>

is the channel number and unit number of the disk spindle you want to add to the system.

ID <number 1-59>

indicates the disk ID number and can be any number from 1 to 59.

SUBSYSTEM <1-8>

Disk devices can be grouped into logical subsystems. Up to 8 logical subsystems can be declared. This element is required.

Indicate the number of the logical subsystem to which this spindle belongs.

DEFAULT

is optional and declares this spindle as a default disk.

If one spindle on a logical subsystem is specified as DEFAULT, all spindles on that subsystem must be declared DEFAULT as well.

More than one logical subsystem can be declared default.

If no subsystem has been declared default, logical subsystem 01 becomes the default subsystem and the residence of the MCP code file. This is not recommended for performance reasons.

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

SHARED <SSP name>

SHARED VIA <SSP name>

Use this option if you are declaring a system in a multi-system shared configuration and want to share disk among the different systems.

Indicate the name of the Shared System Processor (SSP) that controls the concurrent file access between systems. Names are assigned to SSPs with the DLP SSP record.

If the files on this spindle are to be used by programs running on the other systems in the shared system, the spindle must be declared **SHARED** in each system's configuration files. It must also be declared with the same ID number and as part of the same logical subsystem.

If one **DISK** drive on a subsystem is declared shared, all **DISK** spindles on that subsystem must be declared shared.

<beginning address> <ending address>

are optional and can be used to specify the beginning and ending address of the disk spindle, indicating to the MCP the amount of room the spindle has for data. If the entire spindle is to be used, the MCP will automatically determine the proper capacity.

The maximum address ranges possible for various types of disk and LAK (look-alike) pack spindles are as follows:

Table 2-3. DISK/LAK Pack Maximum Address Ranges

SPINDLE TYPE	BEGINNING	ENDING
5N disk (4 DSs)	0	221183
235 pack as LAK	0	969144
206I pack as LAK	0	565729
206S pack as LAK	0	476189
207I pack as LAK	0	998854
207S pack as LAK	0	998854
659I pack as LAK	0	999999
6595 pack as LAK	0	99999
677I pack as LAK	0	999999
677S pack as LAK	0	999999
MD4	0	682259
QWIK disk (15 megabytes)	0	149999

RESERVED

reserves the unit after a cold start or halt/load. The drive remains reserved, unavailable to the system until made ready with the UR (Inhibit or Uninhibit Unit) command. The MCP disk cannot be reserved.

SAVED

puts the drive in a saved state after cold start or halt/load. The MCP disk cannot be saved. See the SV (Save a Peripheral Unit) for more information.

Examples

DISK 6/3 ID 2 SUBSYSTEM 3 RESERVED

adds to the system environment the disk spindle on channel 6, unit number 3. This spindle has a disk ID number of 2 and belongs to logical subsystem 3. After a cold start or halt/load this will be made unavailable to the system until unreserved by a UR command.

DISK 6/4 ID 5 SUBSYSTEM 5 DEFAULT

adds to the system environment the disk spindle on channel 6, unit number 4. This spindle has a disk ID number of 5 and belongs to logical subsystem 5. This disk will be a default disk, and subsystem 5 will be a default subsystem.

See also:

SV (Save a Peripheral Unit) and UR (Inhibit or Uninhibit Unit) commands.

The DISK and DLP SSP records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

DIR (List File Directory)

MCP Control Instruction

Access Level: 5

The DIR command invokes the SYSTEM/COPY intrinsic, which searches file directories of tape, pack, or disk. The command is used to search a directory for one file or one group of files. It can also list all files in the directory. DIR searches tape directories produced by SYSTEM/COPY or user-supplied LOADMP and PACKUP utilities. File names are displayed on an ODT. A printed version of the file list can be created.

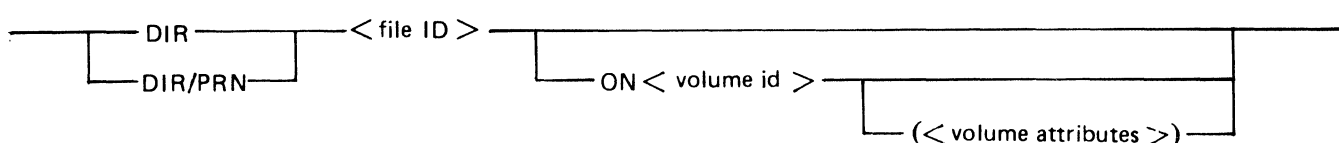


Figure 2-64. DIR Command Syntax

DIR

DIR/PRN

DIR displays the file list on the ODT. DIR/PRN displays the file list on the ODT, and also places the list in a printer backup file for later printing.

If DIR/PRN is performed on a SYSTEM/COPY tape, the printer backup file contains the list of file names, the tape name, the serial number, the creation date, the tape format, the reel number, first file number, and the library level of the tape.

If DIR/PRN is performed on a LOADMP/PACKUP tape, the printer backup file contains only the file list, tape name, the serial number, the creation date, and the tape format.

<file id>

specifies the files in the directory. The equal sign (=) can be used as a masking character to search for a group of files. If one equal sign is entered as the <file id>, all of the files in the directory are listed.

<volume id>

determines which directory will be searched. The volume id can be one of the following choices:

<Tape Name>

searches the directory of a specific magnetic tape.

<Pack Name>

searches the directory of a specific pack.

DISK

searches the directory for all 100-byte media.

PACK

searches directories of all unrestricted packs on the system.

TAPE

searches the directory of a tape with the default name of SYSTEM.

If no <volume id> is entered (that is, if the ON phrase is not used), DISK will be assumed. More details about <volume id> and <volume attributes> are explained in the volume specification syntax of the COPY command earlier in this section.

<volume attributes>

apply to named packs or tapes. The <volume attributes> further define the device entered as <volume id>. If an individual tape name is entered as <volume id>, the <volume attribute> must be TAPE. An individual pack name must be followed by a <volume attribute> of PACK. The <volume attributes> of a SYSTEM/COPY tape can optionally include a VOLUMEINDEX attribute.

See the description and syntax under the COPY (Invoke File Transfer Utility) command for a complete explanation of these attributes.

Uses

DIR can be used on SYSTEM/COPY tapes to determine which reel of a multi-reel volume set contains a particular file. Each reel contains a directory that lists only the files stored on the current and subsequent reels. Analyze the directory of each reel, beginning with the last, until the file is listed. The file is contained in the first reel whose directory shows the file.

Details

The directory on a continuation reel of a multi-reel set will probably not be found at the beginning of the reel; instead, it is located past the end of the file that is continued from the preceding reel. Depending on the length of the continued file, the directory can be some distance into the reel. If a single file requires more than one full reel, it is possible that one or more continuation reels will not have directories.

Examples

DIR = ON TAPEA1 (TAPE)

displays all the files stored on a tape named TAPEA1.

DIR TEST= ON PACK

searches all unrestricted packs, and displays all files whose names begin with the characters TEST.

DIR PR= ON PMDUMP (TAPE, VOLUMEINDEX = 5)

searches the fifth reel of a SYSTEM/COPY tape named PMDUMP and displays all files whose names begin with the characters PR.

DIR/PRN = ON PUBLIC (PACK)

searches a pack named PUBLIC, displays all files stored on that pack, and places the file list into a printer backup file for later printing.

See also:

COPY (Invoke File Transfer Utility) and PD (Print Directory) commands.

DISPLAY/WHATS (Display Information)

Keyboard Input Command

Access Level: 5

The DISPLAY/WHATS command displays code file or disk/pack file information on the ODT. DISPLAY can obtain and display information from the following sources:

- A program code file, the MCP code file, or an MCP-bound intrinsic code file.
- A disk or pack file.
- An absolute disk sector.

The DISPLAY command has three different syntax options, which are shown in Figures 2-65, 2-66, and 2-67.

Displaying a Program Code File, MCP Code File or MCP Intrinsic Code File

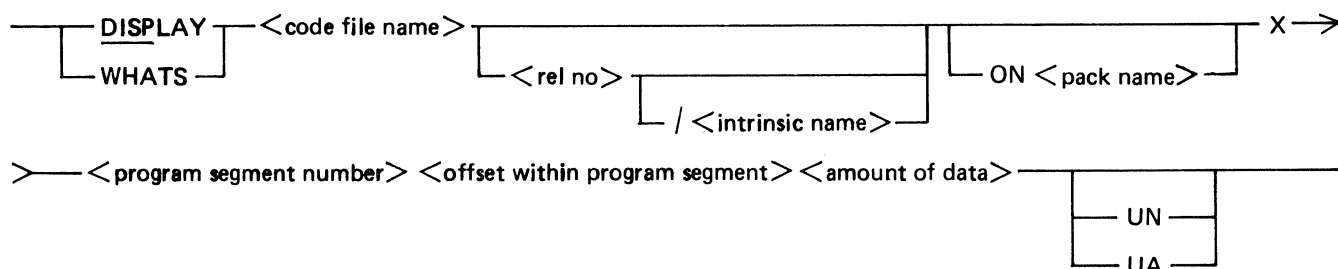


Figure 2-65. DISPLAY (Program, MCP, or Intrinsic Code File)

DISPLAY
DISP
WHATS

are synonymous.

<code file name>

displays a program code file. To display the MCP code file or an MCP intrinsic code file, enter the name of the code file containing the currently running operating system.

<rel no>

refers to the release level of the currently running MCP. Do not enter <rel no> unless the code file to be displayed is either the MCP code file or that of an intrinsic program bound to the MCP. For <rel no> to be valid, the name entered as <code file name> must be the name of the currently running MCP code file.

/ <intrinsic name>

is the name of an intrinsic program bound into the MCP. For <intrinsic name> to be valid, the name entered as <code file name> must be that of the currently running MCP code file.

<pack name>

is the name of the pack where the file or code file is stored. If the ON <pack name> clause is not entered, DISK is assumed. (The generic name PACK is not valid as <pack name>).

<program segment number>

indicates the number of the actual segment within the program where the desired information is located. A valid entry can be any number from 0-999. If the MCP code file is being displayed, this is the environment number (in decimal) of the code area of the desired MCP module or overlay.

<offset within program segment>

is the offset within the program segment where the desired information is located. This offset is expressed in digits.

<amount of data>

is the amount of information to be displayed. The maximum <amount of data> is 30 digits or bytes.

UN

UA

represent the format of the information to be displayed. UA is alphanumeric; the default is UN (numeric).

Displaying a Disk or Pack File

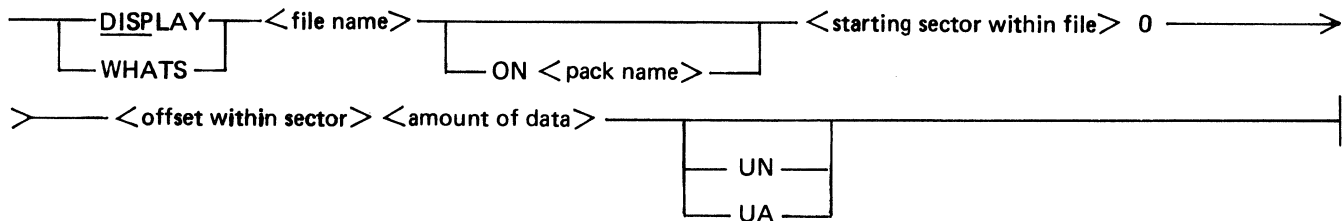


Figure 2-66. DISPLAY (Disk or Pack File)

<file name>

displays information stored in a disk or pack file.

<pack name>

is the name of the pack where the file is stored. If the ON <pack name> clause is not entered, DISK is assumed. (The generic name PACK is not valid as <pack name>).

<starting sector within file>

relative sector within a disk or pack file where the desired information is located. The first sector within the file is numbered zero. Disk sectors are also referred to as segments.

0

separates the starting sector number from the offset number.

<offset within sector>

is the number of digits from the beginning of the disk or pack sector where the desired information is located. The first digit within the sector is numbered zero. Disk sectors are also referred to as segments.

<amount of data>

is the amount of information to be displayed. The maximum <amount of data> is 30 digits or bytes.

UN

UA

represent the format of the information to be displayed. UA is alphanumeric; the default is UN (numeric).

Displaying an Absolute Disk Sector

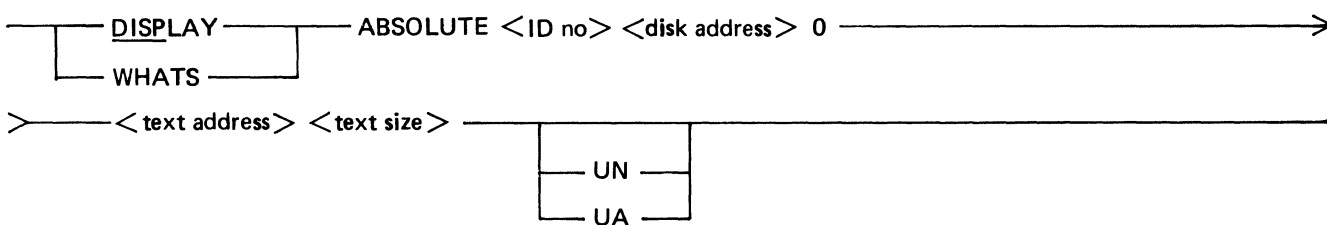


Figure 2-67. DISPLAY (Absolute Disk Sector)

V Series MCP/VS System Software Operations Guide, Volume 2
The System Commands

<ID no>

can be any number from 1 to 255, indicating the logical ID number of the disk.

<disk address>

indicates the number (in decimal) of the absolute disk sector that contains the information to be displayed. This is also called a physical disk address. This address can be 1 to 12 digits long. Disk sectors are also referred to as segments.

<offset from sector address>

indicates the displacement in digits from the beginning of the disk sector being displayed. The first digit within the sector is numbered zero. There are 200 digits in a disk sector. Disk sectors are also referred to as segments.

Uses

The DISPLAY command is most often used in conjunction with the PATCH (Patch File) command. Before patching a given location within a code file or file, display the information stored at that location. This lets you verify that the sector numbers, offsets, and other values to be used in the PATCH command are calculated correctly.

Examples

DISPLAY MONITR X0 20 10 UN

displays information within a program code file named MONITR that is stored on disk. The operating system will display the 10 digits of numeric information that are stored 20 digits from the beginning of the first segment of the program.

DISP ACTR01 ON ACCTNG 1 0 1 10 UN

displays information within a file named ACTR01 stored on a pack named ACCTNG. The operating system will display the 10 digits of numeric information that are stored 1 digit from the beginning of the second sector of the file. (The first sector of the file is numbered zero, as is the first digit within that sector.)

WHATS ABSOLUTE 32 140010 0 20 10 UA

displays information starting at sector address 140010 on the disk drive that has an ID number of 32. The operating system will display the 10 digits of alphanumeric information that are stored 21 digits into the sector address 140010. (The first digit of the sector is numbered zero.)

See Also:

PATCH (Patch File) and SHOW (Display System Information) commands.

DL (DELETE PERIPHERAL DEVICES)

Keyboard Input Message

Access Level: 8

The DL command deletes peripheral devices from the list of peripherals maintained by the system.

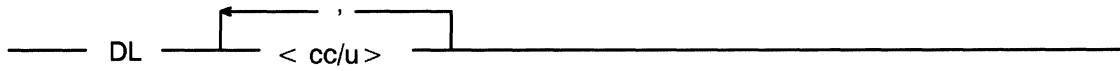


Figure 2-68. DL Command Syntax

<cc/u>

is the channel number and unit number of the device to be deleted. More than one device can be deleted by entering several channel/unit pairs, separated by commas.

Details

You cannot enter DL while the desired peripheral device is assigned to an executing program.

The UNIT (Add Peripheral Devices) command can be used to add a unit back in after it is deleted.

Several records in the system configuration file associate peripheral devices with their respective channel and unit numbers. Applicable records include the DISK record, PACK record, and the various UNIT records.

Examples

DL 6/1

deletes the peripheral device designated as unit number 1 on channel number 6.

DL 4/4, 12/1

deletes the device designated as channel 4, unit 4 and the device designated as channel 12, unit 1.

See Also:

DD (Delete DLPs); DLP (Add DLPs); DISK (Add A Disk); PACK (Add A Pack); SV (Save A Peripheral Unit); UNIT (Add Peripheral Devices); and UR (Inhibit Or Uninhibit Unit) commands.

The DISK, PACK, and UNIT records in the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

DLP (ADD DLPS)

MCP Control Instruction

Access Level: 8

The DLP command adds Data Link Processors (DLPs) to the system environment while the operating system is running. The syntax of the DLP command differs when peripherals are shared between more than one system. The two syntax options are shown in Figures 2-69 and 2-70.

The syntax for the DLP command corresponds to the syntax for the DLP and DLP SSP records used in the system configuration file.

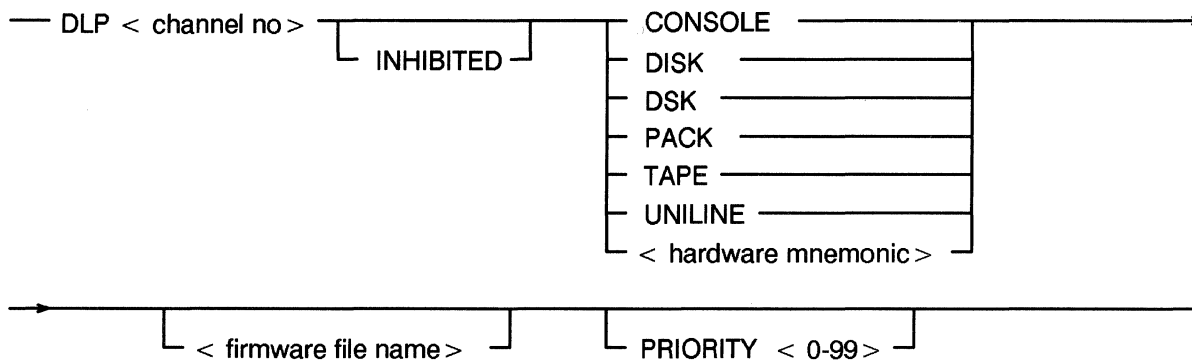


Figure 2-69. DLP Command Syntax (Non-SSP)

<channel no>

is the number of the physical channel where the DLP resides.

INHIBITED

enters the DLP into the system environment in an inhibited state, that is, unavailable for use. (You can use the XC (Inhibit/Uninhibit Channel) command to uninhibit the channel.)

CONSOLE

sets up the channel to serve a Universal Console. If you use this option, do not declare a <firmware file name>.

■ DSK or DISK

declares that the DLP supports disk and pack units (100-byte media and 180-byte media).

■ PACK

■ This indicates that the channel supports diskpack devices.

■ TAPE

■ This indicates that the channel supports tape devices.

GCR, MPE, MTP and MT9 tape devices can be declared using either the keyword TAPE or the hardware mnemonic of the device. Tape devices introduced in the 2.2 and succeeding releases of the operating system are declared using *only* the keyword TAPE. This includes MTC tape devices.

UNILINE

declares a UNILINE DLP.

<hardware mnemonic>

represents the type of DLP being added if it is other than disk, pack, SSP, universal console, or UNILINE DLP. See Table 2-4 for a table of these mnemonics.

Table 2-4. Hardware Mnemonics

CRD	80-column card reader
DCP	Data communications Processor
GCR	GCR magnetic tape drive
IPP	Intelligent Laser Printing System
ISC	Inter-System Connect
MPE	Phase-Encoded (PE) magnetic tape drive
MT9	NonReturn Zero magnetic tape drive
NCP	Network Communications Port
NST	Non-Status device
OCS	Operator Control Station
ODT	Operator Display Terminal
PCH	80-column card punch
PRN	buffered printer
RJE	Remote Job Entry device
S4A	9137 type reader sorter
S4B	9138 type reader sorter
TC5	TC 500 device
TWX	teletypewriter through a UNILINE DLP
TPR	train printer
VDD	B 20 or ET series workstation or a B 9352 or TD 800 series terminal

<firmware file name>

is the name used as the default firmware file name for the DLP being added.

PRIORITY <0-99>

assigns a relative priority to the channel. The value can be any number from 0 to 99, with 99 being the highest priority.

It is recommended that you assign each channel a priority so that when I/Os occur simultaneously, the system knows which one to process first. In such a case, the channel with highest priority gets processed first.

If one or more channels have the same priority, the channel declared first is serviced first.

If channels are not assigned priorities, they are serviced in the order in which they are declared. That is, the channel declared first is serviced first.

Table 2-5 lists the recommended priority settings for the various devices.

Table 2-5. Recommended Channel Priorities

<i>Hardware Type</i>	<i>Priority Range</i>
DCP, ISC, DC-DLP, NCP	96-97
disk and pack	90-95
magnetic tape	80-89
console DLP	70
UNILINE DLP	60-69
printer devices	40-49
card devices	20-29

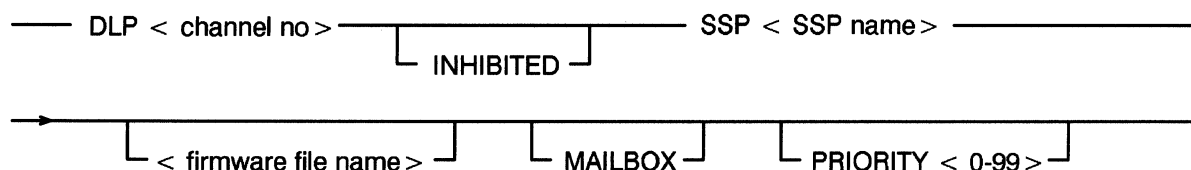


Figure 2-70. DLP Command Syntax (SSP)

<channel no>

indicates the channel the SSP is connected to. It is recommended that each system in a multi-system shared configuration use the same channel number for the same SSP.

INHIBITED

See the description under Figure 2-69.

SSP <SSP name>

declares that the channel serves an SSP DLP in a shared base. The <SSP name> identifies the SSP. It is recommended that all systems in a multi-system shared configuration use the same name for the same SSP. This name will be used in DISK and/or PACK UNIT records that are declared as SHARED.

<firmware file name>

documents the name of the firmware file used by the SSP.

Note that loading firmware to an SSP clears all its lockout entries. Load firmware to an SSP only when all the systems in the configuration are in a preinitialized state; when the MCPs are not running.

MAILBOX

declares that the SSP being added contains the intersystem communication mailbox used to access shared peripheral devices. There can be only one MAILBOX SSP per shared system.

PRIORITY <0-99>

See the description under Figure 2-69.

Examples

DLP 01 DSK HSTLQG PRIORITY 92

defines an access path to a physical disk or pack subsystem on channel 1. The firmware file HSTLQG will be loaded to the controller if an LH (Load Firmware Into A Channel Controller or DCP) command for channel 1 is entered or if the controller is not ready. The priority of this channel is set to 92.

DLP 06 MPE PRIORITY 83

defines a phase-encoded (PE-type) magnetic tape DLP on channel 6, with a priority of 83.

DLP 12 UNILINE USP2LH

defines a UNILINE DLP on channel 12. The firmware file for the UNILINE DLP is USP2LH.

DLP 77 SSP DISKA SSP302 MAILBOX

defines an SSP attached to channel 77. The SSP on this channel controls all shared peripheral devices that include the SSP named DISKA on their system configuration records. The firmware for this SSP is stored in the file SSP302. This SSP contains the intersystem communications mailbox.

See Also

DD (Delete DLP); LH (Load Firmware Into A Channel Controller Or DCP); PACK (Add Pack); UNIT (Add Peripheral Devices); and XC (Inhibit/Uninhibit Channel) commands.

The DLP and DLP SSP records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

DM (DUMP AND CONTINUE)

Keyboard Input Message

Access Level: 1

The DM command produces a memory dump of an executing task and continues execution of the task after the memory dump is complete.

To produce a memory dump of a user task, see Figure 2-71; for a dump of the MCP mix, see Figure 2-72.

Memory Dumps of a User Task

The operating system dumps the memory associated with a user task into a file on disk named \$snnnn, where s indicates the system executing the task and nnnn is a sequential number assigned by the operating system.

Once the memory is dumped, it can be printed and analyzed if you use the PM (Print Memory Dump) command.

— < mix no > DM —————

Figure 2-71. DM Command Syntax (User Task)

■ <mix no>

■ is the mix number of the task to be dumped.

Memory Dumps of the Operating System

The memory dump file of the operating system is always named \$s0001, where s is the number of the system. This file contains the contents of the system's memory at the time the command is entered.

Once the file is dumped, it can be printed and analyzed by using the PM (Print Memory Dump) command.

— 0 DM —————

Figure 2-72. DM Command Syntax (MCP Mix)

DM is the only system command that allows a mix number of 0.

Details

The DUMP option, if set, automatically initiates an MCP dump if the operating system fails. The dump can be directed to tape, disk, or pack. The function of the 0 DM command depends on the DUMP option, as follows:

- If no USE DUMP record was present when the system was cold started, 0 DM is not allowed. An error message is returned if the command is entered.
- If the media specified on the USE DUMP record is TAPE, 0 DM is not allowed. You must use the DMPMEM utility to dump the MCP to tape. An error message is returned if the command is entered.
- If the media specified on the USE DUMP record is DISK or PACK, 0 DM is valid and the operating system creates a memory dump file on the indicated media.

Example

43 DM

produces a memory dump of the task associated with mix number 43.

See also:

DP (Dump And Discontinue) and PM (Print Memory Dump) command.

The USE DUMP record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The DMPANL utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

DP (Dump and Discontinue)

Keyboard Input Message

Access Level: 1

DP specifies that a memory dump of the indicated object program is to be produced and the program discontinued. The DP command is essentially a combination of the DM (Dump And Continue) and DS (Discontinue Program) commands.

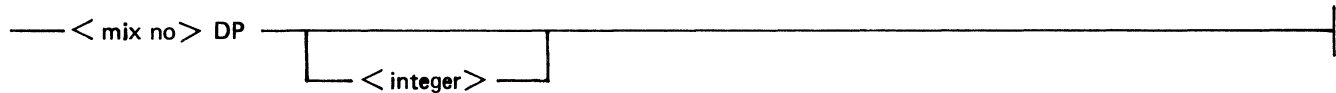


Figure 2-73. DP Command Syntax

<mix no>

is the mix number of the program to be dumped and discontinued. The mix number cannot be 0 or 1, which indicates the MCP, or specify a locked program or one that is already terminating.

<integer>

is entered into the MCP Run Log EOJ record for analysis and can be any value from 00 through 99. Undigits are not permitted. If an <integer> is not specified, a default value of 00 is used.

Details

Once the file is dumped, it can be printed and analyzed by using the PM (Print Memory Dump) command.

Examples

7 DP

dumps and terminates the program associated with mix number 007.

14 DP 30

dumps and terminates the program associated with mix number 014 and enters 30 into the Run log EOJ record.

See also:

DM (Dump And Continue); DS (Discontinue Program); and PM (Print Memory Dump) commands.

DQ (Display System Messages)

Keyboard Input Message

Access Level: 0

The DQ command displays system output messages from the ODT log.

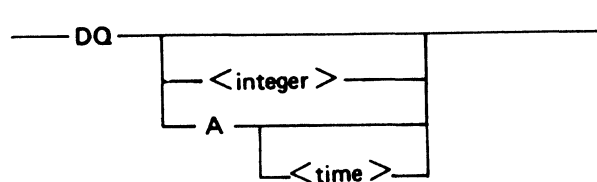


Figure 2-74. DQ Command Syntax

<integer>

specifies the number of previous lines back from the most current line in the queue you want to have displayed. The <integer> can have a value from 1 to 9999, with the default value being the number of lines on the ODT screen.

A

can be used to request that all messages in the ODT log be displayed.

<time>

starts the display of all messages in the ODT log since the specified time (hour and minute). Specify time in 24-hour format.

Examples

DQ 40

displays system messages starting 40 lines back from the most recent entry in the ODT log.

DQ A 1630

displays all messages in the ODT log since 4:30 pm.

See also:

AD (Specify ODT/OCS Screen Parameters); AJ (Display Active Jobs); ES (End Screen Display); LNS (Transfer and Print ODT Log); OL (Display Peripheral Status); TLS (Transfer ODT Log); WJ (Display Waiting Jobs); WS (Display Jobs in Schedule); and WY (Display Job Status) commands.

DR (Date Reset)

Keyboard Input Message

Access Level: 8

DR changes the value of the current system date used by the MCP. This command is not valid if jobs are running in the mix.

— DR <date> —

Figure 2-75. DR Command Syntax

<date>

must be in the form mm/dd/yy, and separating slashes are required. Month, day, and year values must be within valid ranges.

Example

DR 12/16/86

changes the value of the current system date used by the MCP to December 12, 1986.

See Also:

TR (Time Reset); WD (Display Date); and WT (Display Time) commands.

The DATE and TIME records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

DS (Discontinue Program)

Keyboard Input Message

Access Level: 1

The DS command immediately terminates a program.

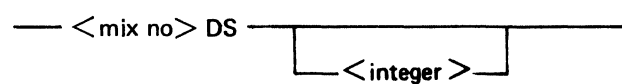


Figure 2-76. DS Command Syntax

<mix no>

is the mix number of the program you want to terminate. The mix number cannot be 0 or 1, nor can it reference a locked program or one in a terminating status.

<integer>

specifies a value to be placed in the Run Log (RLOG=) EOJ record for time analysis purposes. The <integer> can be any value from 00 to 99; undigits (A to F representing hexadecimal 10 to 15) are not permitted.

If the <integer> is not specified, a default value of 00 is used.

Details

You can use the AFTER (Hold Program) command to hold initiation of a program until the completion of another program. If you discontinue a program associated with another through the AFTER command, the following can occur:

- If the AFTR option is set, all jobs scheduled to be executed after the program being discontinued, plus all successors to these jobs, are automatically removed from the schedule.
- If the AFTR option is reset, the scheduled jobs are not disturbed.
- You can manually remove an after-linkage between programs with the RA (Remove After Linkage) command.
- You can manually remove a job from the schedule with the RS (Remove Job From Schedule) command.

CAUTION

If DS is used to terminate a stopped job, the result may be a temporary priority crashout of other jobs, or the job may be marked "to be resumed" if insufficient memory is available.

Example

101 DS

terminates the program associated with mix number 101.

35 DS 12

terminates the program associated with mix number 035 and specifies 12 to be placed in the RLOG EOJ record.

See also:

AFTER (Hold Program); DP (Dump And Discontinue); QT (Quit Program Operation); RA (Remove After Linkage); RS (Remove Job From Schedule); ST (Suspend Program Processing); and STOP (Suspend Program Processing) commands.

The USE AFTR record in *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

DUMP (Copy Files To Library Tape)

MCP Control Instruction

Access Level: 5

The DUMP command copies disk or pack files to a library tape without removing the files.

It is recommended that you use the COPY command instead of the DUMP command. The DUMP command will become unavailable at a future release.

The DUMP command starts a program called LOADMP, if the files are to be copied from disk, or a program called PACKUP, if files are to be copied from pack. LOADMP and PACKUP must be written by the user; Unisys does not supply them. If the appropriate program is not present on disk, the DUMP command is ignored.

See Appendix B for DUMP syntax, if you have user-coded LOADMP and/or PACKUP utilities.

See Also:

COPY (Invoke File Transfer Utility) command. Appendix B.

DX (Display DLPs Attached To Exchange)

Keyboard Input Message

Access Level: 8

This command lists the DLPs that are attached to an exchange.

— DX <exchange number> —

Figure 2-77. DX Command Syntax

<exchange number>

indicates the exchange for which you want to list the DLPs.

The system displays the same information about each DLP that is produced by the SHOW (Display System Information) command. For example:

Channel	Priority	State
004	90	Available
014	90	Available

where the first channel is the primary channel (that is, the exchange number), the following channels listed are alternate channels, and the State is one of the following: Available, Busy, Inhibited.

If you enter an invalid exchange number, the system responds:

**KBD IGNORED: INV EXCH

See also:

OL (Display Peripheral Status) and SHOW (Display System Information) commands.

EBCDIC (Mark Beginning Of Card Data File)

MCP Control Instruction

Access Level: 0

The EBCDIC command indicates the beginning of a punched card data file.

Whenever the EBCDIC command is encountered in the card reader, it causes an End-Of-File (EOF) condition on any previous file, and the unit is marked by the MCP as containing the new file.

EBCDIC is valid only from an actual or pseudo card deck.

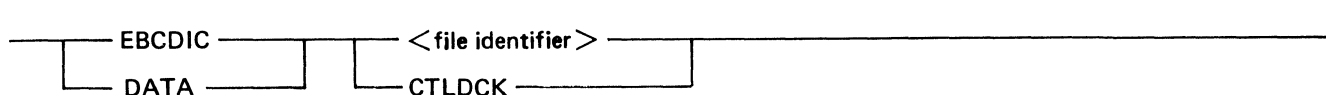


Figure 2-78. EBCDIC Command Syntax

EBCDIC

DATA

signify that subsequent input is in EBCDIC format. EBCDIC stands for Extended Binary Coded Decimal Interchange Code, a coded character set in which each character is represented by an eight-bit code.

<file identifier>

is the name of the file to follow.

Any information following the <file identifier> on the same punched card is ignored.

CTLDCCK

is a reserved file identifier that allows valid control records to be included in the file. CTLDCCK is the name of the physical card deck used by the LDCNTL utility to create a pseudo card deck.

NOTE

If the <file identifier> is CTLDCCK, no EOF action is taken until the ?ENDCTL control instruction is encountered. This is primarily intended for use by the utility program LDCNTL for building pseudo card decks.

Examples

COMPILE PROG COBOL9; EBCDIC CARD

compiles the COBOL program PROG using the COBOL9 compiler. Then, this statement specifies the beginning of a punched card data file called CARD. CARD is in EBCDIC format.

DATA INPUT

specifies the beginning of a punched card data file called INPUT. The file is in EBCDIC format.

See also:

BCL (Mark Beginning Of Card Data File); BINARY (Read Binary Card Deck); DATA/DATAB (Mark Beginning Of Card Data File); and END/ENDCTL (Mark End of Card File or Control Instructions) commands.

ED (Enter DEBUG Session)

Keyboard Input Message

Access Level: 8

The ED command allows you to go back to a specified DEBUG session from the ODT.

ED <session no>

Figure 2-79. ED Command Syntax

<session no>

is a unique four-digit number, prominently displayed within each session.

Example

ED 1387

causes you to re-enter DEBUG session number 1387.

See also:

DEBUG (Attach Program To DEBUG Session), ID (Invoke DEBUG Session), and QD (Query DEBUG Sessions) commands.

The DEBUG utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

END/ENDCTL (Mark End of Card File or Control Instructions)

MCP Control Instruction

Access Level: 0

The END command indicates the end of a card data file or sequence of control instructions. This command is valid only from an actual or pseudo card deck.

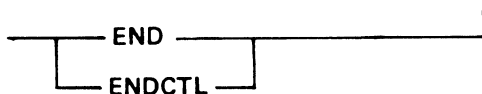


Figure 2-80. END/ENDCTL Command Syntax

END
ENDCTL

indicate the End-Of-File (EOF) for a punched card file or the end of a series of MCP control instructions. The END card is always the last card of a file, a series of files, or a series of control instructions and must not be followed by any further data or control instructions for that file or program.

ENDCTL is the only command that provides EOF action on any card file having the reserved <file identifier> CTLDCK. Note that CTLDCK cannot be a BINARY file.

Details

When an END card is sensed by the LDCNTL utility, the pseudo deck (disk file) is closed, and the deck is given a unique numeric designation for operator interface. The card reader remains attached to the program LDCNTL until the special end card ?ENDCTL is reached, causing LDCNTL to terminate.

See also:

BCL (Mark Beginning Of Card Data File); BINARY (Read Binary Card Deck) DATA/DATAB (Mark Beginning Of Card Data File); and EBCDIC (Mark Beginning of Card Data File) commands.

ENDAT (End Job Transfer)

MCP Control Instruction

Access Level: 0

The ENDAT command terminates the job transfer previously initiated by an AT (Initiate BNA Job Transfer) MCP Control Instruction.

The ENDAT command is valid only from an actual or pseudo card deck.

— ENDAT —|

Figure 2-81. ENDAT Command Syntax

See also:

AT (Initiate BNA Job Transfer) MCP control instruction for a complete description of ENDAT.

ENDUSER (Mark End Of Job Stream)

MCP Control Instruction

Access Level: 0

The ENDUSER command delimits the end of a job stream that has been assigned a <usercode>/<password>/<charge number> access combination by a BEGINUSER command.

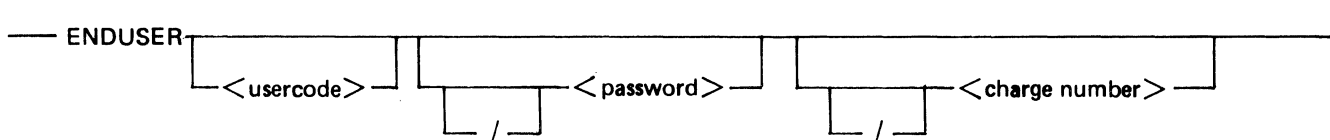


Figure 2-82. ENDUSER Command Syntax

<usercode>

refers to a code that identifies a user to the system.

<password>

refers a group of characters by which the system determines the legality of a user.

<charge no>

refers to a number used to assign charges for system use.

NOTE

The <usercode>, <password>, and <charge number> are all optional. Slashes are required only if leading parameters are omitted (for example, ENDUSER //30442).

Details

The <usercode>/<password>/<charge number> access combination specified by the ENDUSER command must match that of the previous BEGINUSER (Mark Beginning of Job Stream) command. If these access combinations match, then the access combination from the BEGINUSER or LI (Log In) command prior to the last BEGINUSER command becomes the active combination for the device.

BEGINUSER and ENDUSER commands can be nested. However, the access combination specified on the ENDUSER instruction must always match the access combination of the last BEGINUSER instruction.

Example

The following example shows a series of nested BEGINUSER and ENDUSER statements.

BEGINUSER USR1/PASS1/99999	marks the beginning of the job stream and assigns any function requests for that job stream to usercode USR1, password PASS1, and charge number 99999.
EX PROGA	executes a library program called PROGA.
EX PROGB	executes a library program called PROGB.
BEGINUSER //88888	marks the beginning of the job stream and assigns any function requests for that job stream to charge number 88888.
EX PROGD	executes a library program called PROGD.
?SPO LNR	sends a LNR (Transfer and Print Log) command to the MCP through a card reader to close the current system log file and directs logging information to the Run Log and starts the log analysis program.
ENDUSER //88888	marks the end of the job stream assigned to charge number 88888.
ENDUSER USR1/PASS1/99999	marks the end of the job stream assigned to usercode USR1, password PASS1, and charge number 99999.

See also:

BEGINUSER (Mark Beginning of Job Stream) and USER (Assign Access Code) commands.

The *V Series System Software Security Installation and Operations Guide* for information about <usercode>, <password>, and <charge no>.

ES (End Screen Display)

Keyboard Input Message

Access Level: 0

When the MCP response to a system command requesting information contains more data than can be displayed on the ODT or OCS at one time, the MCP displays one screen page and the "MORE DATA" message. The ES command causes the MCP to purge the information not being displayed and to return to automatic display mode.

— ES —

Figure 2-83. ES Command Syntax

Details

This command is valid only from an ODT or OCS.

If you enter an ES command during normal processing, the MCP updates the automatic display of system status.

See also:

AD (Specify ODT/OCS Screen Parameters) command.

ET (End TRAK)

Keyboard Input Message

Access Level: 8

The ET command terminates a TRAK facility tracking session.

TRAK is a real-time debugging tool that records various MCP data structures or events. TRAK entries are stored in a TRAK buffer in memory and can be analyzed by DMPANL, the dump analysis utility.

TRAK tracking is initiated by the BT (Begin TRAK) command.

— ET —

Figure 2-84. ET Command Syntax

Details

After using the ET command, the TRAK buffer can be printed and analyzed by first creating a dump file through a 0 DM command followed by a "PM 1 TRAK" command. See DM (Dump and Continue) and PM (Print Memory Dump) in this section, for more information.

The TRAK option can be reset with the RO (Reset System Options) command.

See Also:

BT (Begin TRAK); DM (Dump and Continue); PM (Print Memory Dump); RO (Reset System Options); and SO (Set System Options) commands.

The DMPANL utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

EXECUTE (Execute Library Program)

MCP Control Instruction

Access Level: 1

The EXECUTE command is used to execute a library program on disk or pack.

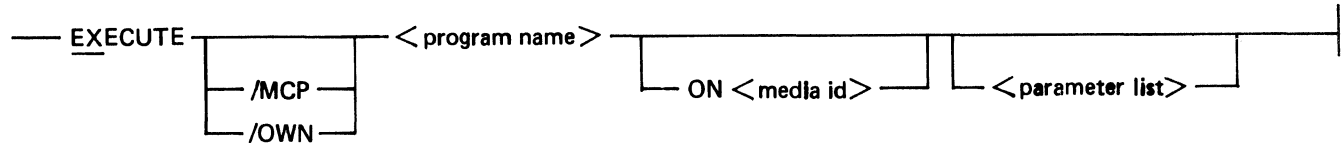


Figure 2-85. EXECUTE Command Syntax

EXECUTE

EX

are synonymous.

/MCP

specifies that the bound MCP intrinsic of the name specified is the program to be executed. A user-coded intrinsic of the same name is not executed.

If you specify this option, the ON <media id> option (if included) must specify DISK or not be used.

/OWN

specifies that only the user-coded intrinsic with the correct name can be used.

If the /MCP or /OWN option is omitted, the user-coded intrinsic will be used if present on disk. If not, then the MCP intrinsic will be executed.

<program name>

is the name of the program to be executed. It must be a permanent file on disk or on the named pack (or pack family if more than one pack is present with the appropriate name).

<media id>

specifies to the MCP where the program resides. The <media id> can be one of the following:

DISK

directs the MCP to look for the program to be executed only on 100-byte media. If the program is not found, an appropriate message will be displayed by the MCP.

PACK

directs the MCP to search all of the system's unrestricted packs for the program to be executed. If the program is not found, an appropriate message will be displayed. The program must be in 180-byte code file format, either by copying the program from disk to pack using the PCOPY utility, or by directing the generating compiler to create such a code file. See COMPILE (Compile Program), in this section.

A pack family name

directs the MCP to search for the program to be executed on that pack family name only. If not found, an error message will be displayed.

If no <media-ID> is specified, the MCP looks first on disk, then on unrestricted pack.

<parameter list>

are up to three parameters that can be inserted into a program when the program is started.

The <parameter list> option appears in the syntax of the DEBUG (Debug Session), EXECUTE and COMPILE (Compile Program) commands. These parameters allow you to enter Boolean, string, and integer values, which can be used by the program when the program is being compiled or executed.

The syntax for <parameter list> is shown in Figure 2-86.

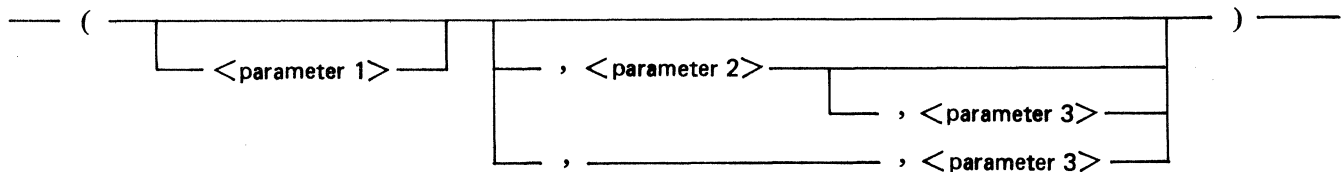


Figure 2-86. Syntax For <parameter list>

The rules for <parameter list> resemble those for parameters in the Work Flow Language (WFL) RUN command.

Integer parameters:

- contain unsigned numeric data up to eight digits in length (8 UN)
- have values ranging from 0-F
- are right-justified and zero-filled in the high-order digits if less than the maximum length
- will cause a syntax error if greater than the maximum length

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

String parameters:

- contain alphanumeric data up to six characters in length (6 UA)
- must be within quotes. To pass a quote as a parameter, a double quote must appear in the parameter string (for example, EXECUTE MYPROG ON PACK (,"TRY""XX").
- may be composed of any EBCDIC characters
- are left-justified
- are blank-filled in the low-order bytes if less than the maximum length
- will be truncated in the low-order bytes if greater than the maximum length

<parameter 1>

can be an integer up to eight digits in length. It lets you set switches at program address 0. For example, EX MYPROG (10100000).

<parameter 2>

<parameter 3>

can be eight-digit integers or six-character strings. <parameter 2> is inserted into the code file at base-relative address 8; <parameter 3> at base-relative address 20. For example, EXECUTE /OWN MYPROG (,12345678) or EXECUTE /OWN MYPROG (101,"TOSTEST").

When either <parameter 2> or <parameter 3> are specified, you must have either a value or at least a comma in the preceding parameters. For example, EX MYPROG /OWN (,,"TAPE").

A null entry is allowed. For example, EXECUTE /OWN MYPROG (,0,"ABC"" ") does not change the switches and inserts 00000000 at address 8 and the string "ABC"" " at address 20.

Details

If you do not provide a <media id>, the MCP will first search all 100-byte disks for the program. If not found in the disk directory, a search of all non-restricted packs is performed. If the program is not found in the disk directory or in any of the unrestricted packs, an appropriate message is displayed.

You can reduce the extensive search that may result from not using the ON <media id> clause by including a CONTROL CODEPACK record in the system configuration file. Including the CONTROL CODEPACK <pack id> record forces the MCP to look for the program on the specified <pack id> only, if the program was not found on disk.

Slash Parameters

The <parameter list> option replaces former MCP slash parameters. MCP/VS 2.0 continues to allow slash parameters. However, when they are used, a warning will be displayed on the ODT and written to the ODT log stating that slash parameters will be deimplemented at the next MCP/VS release.

Note also that the VALUE (Insert Value into a Program) cannot be associated with an EXECUTE command if you specify <parameter list>. The INSERT (Insert Data Before Execution) command, however, can still be used.

If you want to use slash parameters instead of <parameter list>, use the syntax shown in Figure 2-87.

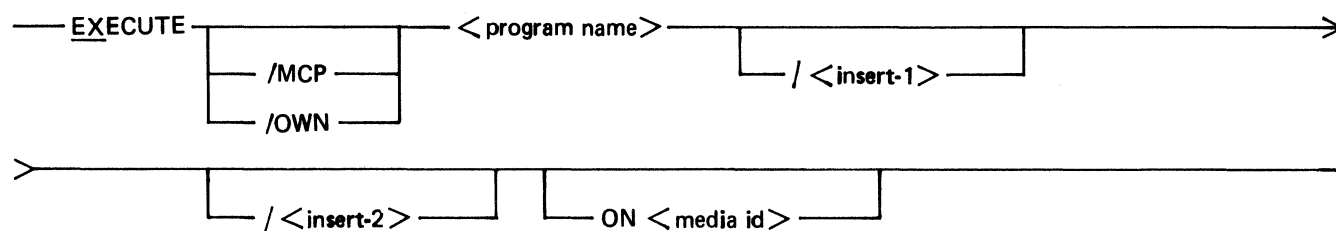


Figure 2-87. EXECUTE Command With Slash Parameters Syntax

/<insert-1>

/<insert-2>

are 1- to 6-byte alphanumeric parameters that can be used for various compilers.

The first parameter, padded with trailing spaces to total six bytes, is placed into the program at base-relative address 8.

The second parameter, if present and padded to six bytes, is inserted at base-relative address 20.

The area where these parameters are inserted is the same area as the index registers. Therefore, the parameters must be interrogated or saved before using the index registers for subscripting, indexing, or procedure entry, or any other purposes.

Examples

EX BIOG ON PACK; PR 5

executes the library program BIOG, which resides on PACK, and then assigns it a memory, processing and I/O, and schedule priority of 5.

EXECUTE /MCP AUDIT (,"TEST10")

executes the library program AUDIT, an MCP intrinsic that resides on disk, doesn't set any switches, and inserts "TEST10" at base-relative address 8.

EX MDCOPV; VA 1 = 1000000

executes the MDCOPV utility, which provides maintenance for flexible disks, and sets all switches or SW 8 to 1.

See also:

COMPILE (Compile Program) command.

The CONTROL CODEPACK record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The PCOPY utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

FI (LIST FILES IN USE BY A PROGRAM)

Keyboard Input Message

Access Level: 0

The FI command will produce a list of all files in use by a program.

_____ < mix no > FI _____

Figure 2-88. FI Command Syntax

<mix no>

is the mix number of the program whose files you want to list.

Example

38 FI

lists all files in use by the program associated with mix number 038. A sample response to this input is as follows:

FILES IN USE BY @20298/PBDPRN = 038

File	Hdw	cc/uu	I/O	I/O Count
@20298	DSK	15/01	I	123
SHORT/DASCIO	PRN	13/00	O	1254

As many response lines as necessary will be produced.

See also:

BF, BFP (List Backup Files) command.

FILE (EQUATE FILE-NAME WITH FILE-IDENTIFIER OR MEDIUM)

MCP Control Instruction

Access Level: 3

The FILE command, an MCP label equation feature, allows a change at execution of the file-identifier and, in some cases, of the hardware medium of a specified file.

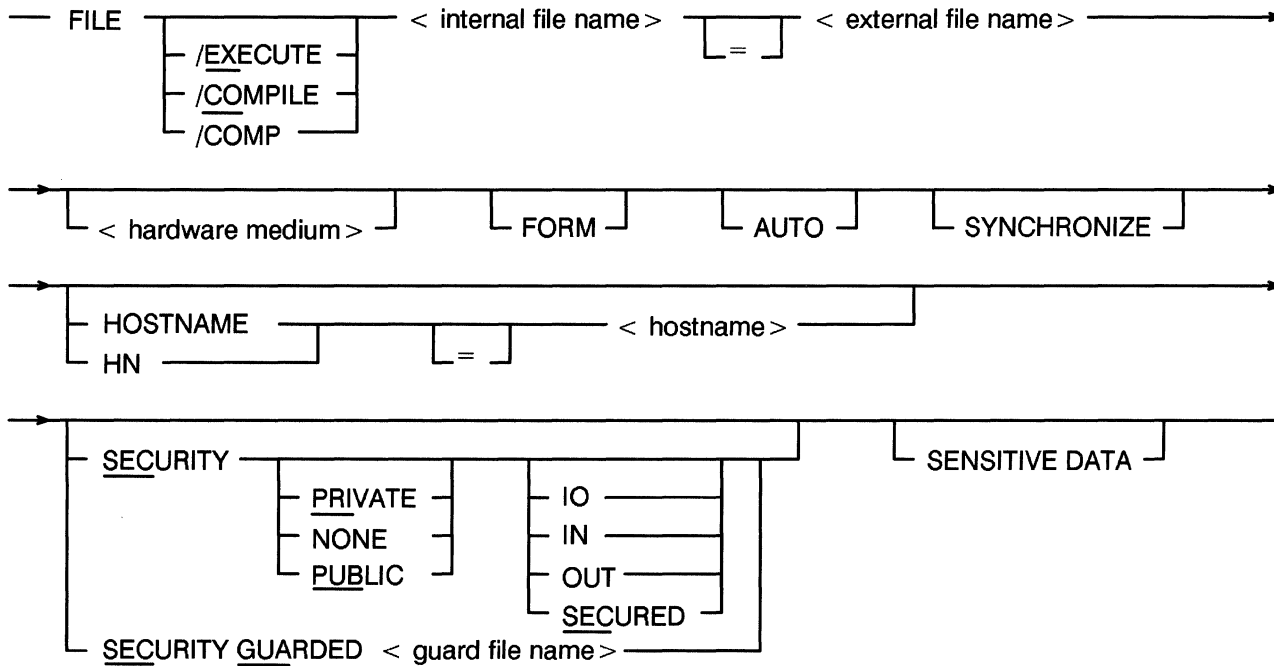


Figure 2-89. FILE Command Syntax

The FILE command equates an <internal file name> as used in a given source program with a specified <external file name>, and optionally with a <hardware medium>. The new file name or <hardware medium> will be used at execution time. A PORT file cannot be specified in a FILE statement.

EXECUTE
EX

are synonymous.

COMPILE
CMP

are synonymous.

/EXECUTE
/COMPILE

can be used to distinguish between the COMPILE and subsequent EXECUTE phases of a compile-and-go operation. If this option is omitted on a compile-and-go, the FILE command is associated with the compiler. This option can be used only on compile-and-go operations.

<internal file name>

is the source language <internal file name> of the file to be label-equated. If the source language <internal file name> consists of more than six characters, only the first six characters can appear in the FILE command. If the <internal file name> contains special characters (for example, PRNT-F), the <internal file name> must be enclosed in parentheses; for example: (PRNT-F).

<external file name>

is the external file label to be used instead of the <internal file name> specified in the source program. For media permitting a <multifile-id>, a <multifile-id>/ <file id> entry can be used.

<hardware medium>

designates an alternate <hardware medium>, which can be any of those listed in Table 2-8.

Table 2-8. Hardware Media and Mnemonics

<i>Mnemonic</i>	<i>Hardware Medium</i>
CRD	Card reader
DISK	Head-per-track disk (100-byte media)
DSK	Head-per-track disk (100-byte media)
DPK	Pack (180-byte media)
GCR	Magnetic tape, GCR only (output)
MTP	Magnetic tape (any)
MT9	Magnetic tape, 9-channel only (output)
MTC	Magnetic cartridge tape (output)
MPE	Magnetic tape, phase-encoded only (output)
PACK	Pack (180-byte media)
PRN	Line printer
PRO	Line printer only
PBT	Printer backup tape
PBD	Printer backup disk
PBP	Printer backup pack
PBK	Printer backup, tape, disk, or pack
PCH	Card punch
PCO	Card punch only
PCD	Card punch backup disk
PCP	Card punch backup pack
QWK	QWIK disk

Table 2-9 shows the allowable file equates across hardware types:

Table 2-9. Allowable File Equates Across Hardware Types

<i>TO FROM</i>	DSK	DPK	MTP	PRN	CRD	PCH	QWK
DSK		OK	NO	NO	OK	NO	
DPK	OK		OK	OK	OK	OK	OK
MTP	OK	OK		OK	OK	OK	NO
PRN	OK	OK	OK		NO	OK	NO
CRD	OK	OK	OK	NO		NO	OK
PCH	OK	OK	OK	OK	NO		NO
QWK	OK	NO	NO	OK	NO		

The hardware usage must be compatible; for example, output tape cannot become CRD.

DISK output cannot be file-equated to anything except pack. DISK input can be file equated to CRD or pack.

DCOM files cannot be file-equated to any other hardware devices, including other DCOM devices.

SYNCHRONIZE

indicates that data is transferred between host and tape drive in an unbuffered mode. The default mode is asynchronous (buffered) which maximizes tape drive performance.

FORM

when following any of the printer or punch options, indicates that special forms are required.

HOSTNAME

HN

are synonymous, and cause the indicated file to be opened on the processor for which this is the hostname. This requires that both the originating processor and the remote processor be included in a BNA network.

Security Attributes

The following attributes apply to output files only.

SECURITY

specifies that security attributes are going to be applied to this file.

PRIVATE

specifies that the file can be accessed only by the creator of the file in the manner specified by the SECURITYUSE attribute. (For a description of the SECURITYUSE attribute, see the *V Series MCP/VS Programming Reference Manual*.) PRIVATE is the default SECURITYTYPE attribute.

NONE

specifies that no security attributes are to be applied to this file.

PUBLIC

specifies that the file can be accessed by all users in the manner specified by the SECURITYUSE attribute.

IO

specifies that the SECURITYUSE attribute will allow READ/WRITE access to the file.

IN

specifies that only READ access is allowed to the file.

OUT

specifies that only WRITE access is allowed to the file.

SECURED

specifies that the file is a program and can be executed only if it passes the MCP requirements for a program. It cannot be opened as a file by a user program.

SENSITIVEDATA

specifies that all records of the file are to be overwritten with random data before the file is returned to the available disk pool. SENSITIVEDATA will be ignored if the file is declared as PUBLIC.

SECURITY GUARDED

specifies that another file (guard-file-name) must be accessed before a user can be given permission to use the file. The SECURITYUSE attribute for a GUARDED file is stored in the GUARD file.

Examples:

EXECUTE A; FILE CARDFL = CARDX

executes library program A; causes the MCP to equate the internal file name CARDFL with the external file name CARDX.

EXECUTE B; FILE PRINTR = SLPRNT PBD FORM

equates the file PRINTR with a printer backup disk file called SLPRNT, requiring special forms.

COMPILE C COBOL9; FILE CARD X DSK; FILE/EX TAP TP1/TP

contains two file-equate statements. The second statement indicates the EXECUTE phase of a compile-and-go operation, and equates the file name TAP with the file name TP1/TP.

Automatic Printer Assignment

Through the use of the FILE statement or by setting the FORMs flag for the print file, the MCP will allow automatic routing of printer files to a specific printer. The automatic routing feature should be applied when the normal state of the printer is SAVED. A printer can be kept in a SAVED state by SVing the printer and then <mix no> FMing all print files to the printer or by using the FORM option of the FILE command to maintain the SAVED status after use.

Automatic routing applies to train printers only and is set up by the UNIT record for the printer. The UNIT record syntax for a printer is shown in Figure 2-90.

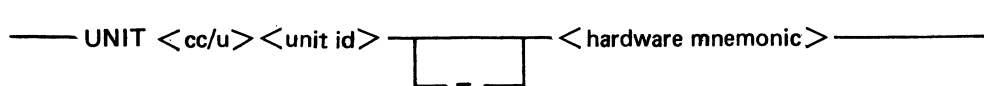


Figure 2-90. UNIT Record Syntax (Printer)

<cc/u>

is the channel and unit number for the printer.

<unit id>

is the default train-id (translate table) to be used for a train printer.

The <unit id> field is used for printer identification when implementing automatic routing. The <unit id> is an identifier (one to six digits) used to make a specific printer unique. When a print file that has FORMs specified is to be printed, the MCP looks at the multifile-id and assigns the print file to the printer whose first four characters of <unit id> are the same as the first four characters of the print file's multifile-id. For example, a print file with a multifile-id of SHORXZ would be assigned to a printer with a <unit id> of SHORT.

<hardware mnemonic>

is the hardware type (PRN, TPR or IPP).

FILE <internal id> _____ <multifile id> / <file id> _____
 └───┬───┘ └── PRO ─┘ └── FORM ─┘ └── AUTO ─┘

<internal id>
is the name of the print file to be printed. The internal id of the print file for PBDPRN is PRINT.

<multifile id>
is the name of the file whose first four characters are the same as the first four characters of the <unit id> for the printer.

<file id>
is the external id of the print file.

PRO
directs the file to the printer only, not to backup, and can be used when file equating with a PRINT or PBD command. PRO must be specified before FORM. This option cannot be used to send a file to an image printer (IPP).

FORM
assigns special forms to a print file that did not have forms assigned programmatically. FORM must be specified for automatic printer assignment. It will leave the printer in a SAVED state.

AUTO
specifies that the APBD option has been set, and that the file is to be printed as soon as it has been created. If APBD = 1, AUTO must be used on the FILE record to automatically schedule the printing of the file. PBD + AUTO + APBD = AUTO print file.

Use the free-standing **LOADER** program to create the train printer translate tables to provide a name that corresponds to <unit id> on the **UNIT** record. You could also copy an existing file and rename it.

The **LOADER** program must be run for each form that is to be maintained on a saved printer.

If all printers on a system have the same train (96-character), **LOADER** has to be run once. If the printers are going to have the same form loaded, only the single translate table needs to be maintained. If the printers are going to have different forms, **DMPALL** or **SYSTEM/COPY** could be used to duplicate the translate table and change the name.

If multiple printers have the same <unit id>, the MCP will assign the print file to the printer that is declared first.

LOADER cannot be used to download firmware to an image printer (IPP). Firmware is loaded onto an image printer by using the LDHOST utility.

Multiple printers can use bound train tables simultaneously.

The automatic routing can be used in an environment where forms are changed frequently. To do so, declare the printer without a train id and load FORMs. Leave the printer not-ready. Transmit: RY cc/u <file-id>; SV cc/u. Manually RYing the printer will save and lose FORM id at end of run. You could also use the FM (Direct File To Device — Special Forms Response) command and indicate a new file-id.

If the printer is a drum printer, the execution of the LOADER program is unnecessary. Only a <unit id> needs to be put on the UNIT record.

Because FORM and MFID are required, the printer must be saved only at halt/load but not made ready with an RY (Ready Peripheral Device) command. A form request will direct to an RYed form printer with a matching UNIT ID, but other NO FORM prints will use it too.

File equate is not necessary if FORM and MFID are specified in the program.

File equate for a program should normally specify PBD/PBP/PBK. When the file is created in this manner, a normal PRINT (without a file equate command) can be done.

FILE Statement Examples

Consider the following printer configuration:

UNIT 07/0 COMPLE PRN SAVED	standard drum printer
UNIT 32/0 LONGFM TPR SAVED	96 character train
UNIT 62/0 SHORTY TPR SAVED	96 character train

Example 1

```
COMPILE LGAD01 COBOL9 LIBRARY
FILE LINE COMPLE/LGADO1 FORM
DATA CARD
END
```

This example will change the file id of this print file from LINE to COMPLE/LGAD01 and set the FORMs option for the file. When the file is printed, it will automatically go to printer 7/0. If AUTO had been specified and the APBD option were on, the file would have printed automatically after it was created.

Example 2

```
EXECUTE LGAD01  
FILE LOGRPT LONGFM/LGAD01 FORM AUTO  
END
```

This example assigns a name of LONGFM/LGAD01 to the print file LOGRPT and sets the FORMs option for the file. AUTO specifies that the file be printed automatically on printer 32/0 after it is created.

NOTE

If the APBD option is not set, the file will not be printed automatically, and a normal PRINT command will have to be issued to the file.

See Also

FM (Direct File To Device — Special Forms Response) and RY (Ready Peripheral Device) commands.
The SECURITYUSE attribute in the *V Series MCP/VS Programming Reference Manual*.
The UNIT and USE APBD records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

FM (Direct File To Device — Special Forms Response)

Keyboard Input Message

Access Level: 7

The FM command lets you reply to the forms-required MCP message: "<hardware mnemonic> REQD <file name> <job name> = <mix no>".

The FM command is identical to the OU (Direct File To Device) command except that the device specified by the <cc/u> can be in a saved status, and the device is left in a saved status when the output process is completed.

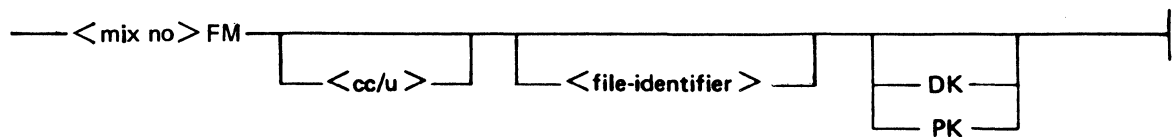


Figure 2-92. FM Command Syntax

<mix no>

must specify the mix number of a program that requires an output printer or punch device.

<cc/u>

must designate the unit to be used for the subject file and must be a valid peripheral type for the file in question.

<file identifier>

can be used to specify which translate file is to be loaded into the control, when the <cc/u> refers to a train printer.

DK

PK

will force the print file to disk backup or pack backup, respectively.

Examples

2 FM 8/0

directs the output of a program with a mix number of 2 to an output device with a cc/u number of 8/0.

12 FM DK

directs the output of a program with a mix number of 12 to a disk printer backup or file.

10 FM 16/0 A123

directs the output of a program with a mix number of 10 to a train printer whose channel number is 16, unit number is 0, and whose translator file is A123.

See Also

OU (Direct File To Device) command.

FN (List File Names)

Keyboard Input Message

Access Level: 3

The FN command produces a list of a program's internal file names and their corresponding external file names. Therefore, the operator or programmer can quickly determine the appropriate file name to use in a FILE equate statement.

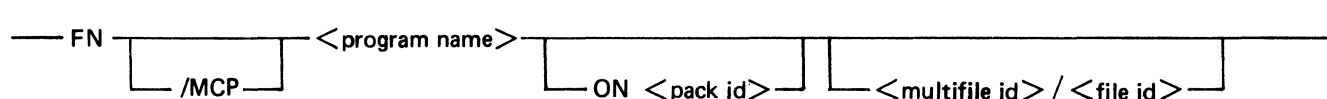


Figure 2-93. FN Command Syntax

/MCP

indicates that the program is bound to the MCP. This option is invalid if the ON <pack id> syntax is used.

<program name>

is the name of the program.

ON <pack id>

indicates that the program is resident on a pack that has a label specified by <pack id>.

<multifile-id>/<file-id>

specifies the external id of the file as displayed on the ODT if a NO FILE condition existed. The syntax is identical to that of a FILE statement when label equating a tape or pack file. The <file-id> can be masked with a single equal (=) sign. Masking is not permitted in the <multifile-id>.

Examples

FN COPY

lists the internal and external file names of the program COPY.

FN/MCP COPY

lists the internal and external file names the MCP-bound program COPY.

FP (Flush Program)

Keyboard Input Message

Access Level: 1

The FP command directs the MCP to purge all I/O requests for a terminating program when peripheral hardware malfunctions are inhibiting the termination. The program normally awaits the completion of processing on an output file after a DP (Dump and Discontinue), DS (Discontinue Program), or STOP (Suspend Program Processing) command.

— <mix no> FP —

Figure 2-94. FP Command Syntax

<mix no>

must refer to a terminating program and cannot be 0 or 1.

CAUTION

Use this command cautiously because all I/O operations pending for the specified mix number at the time of the request are lost when FP is used.

Example:

7 FP

purges all I/O requests for the terminating program associated with mix number 007.

See also:

DP (Dump and Discontinue); DS (Discontinue Program); and STOP (Stop a Program Temporarily) commands.

FR (Designate Final Reel)

Keyboard Input Message

Access Level: 1

The FR command lets you respond to a magnetic-tape or paper-tape file-required message from the MCP by indicating the input reel just read as the final reel of the file. This command is used primarily for unlabeled files, the end of which cannot be recognized by the MCP.

— <mix no> FR —|

Figure 2-95. FR Command Syntax

<mix no>

represents the mix number of the file.

Details

When a program declares a tape to be labeled, the FR command is allowed if you use the UL (Assign Unlabeled File) command to designate the unit on which a particular unlabeled tape file is located. However, FR is not allowed if you use the IL (Assign Labeled File) command, which designates the location of a labeled file.

Example

13 FR

the input reel just read for the file associated with mix number 013 is designated the final reel of the file.

See also:

IL (Assign Labeled File) and UL (Assign Unlabeled File) commands.

GENERATE (Schedule DMPALL For Execution)

MCP Control Instruction

Access Level: 1

The GENERATE command schedules the program DMPALL for execution, passing specified parameters to that program through a disk file.

The GENERATE function of DMPALL generates an Assembler Language program to reproduce a data file from one hardware type to another hardware type, and optionally change file-id, record length, blocking factor, or parity.

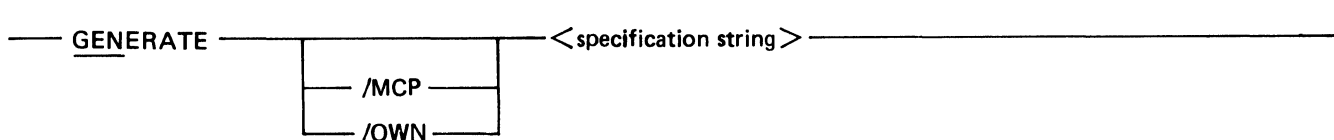


Figure 2-96. GENERATE Command Syntax

GENERATE

GEN

are synonymous.

/MCP

/OWN

specify either the bound MCP-intrinsic DMPALL or a user-provided DMPALL

If /MCP is specified, the bound, MCP-intrinsic DMPALL is scheduled regardless of the presence on disk of a user-provided DMPALL.

The /OWN option can be used to prevent the scheduling of the bound, MCP-intrinsic DMPALL when a user-provided DMPALL is required but is not on disk.

If this option is omitted, the GENERATE command causes the MCP to search the disk directory for a user-provided program called DMPALL. If a user-provided DMPALL is resident on disk, it is scheduled for execution; otherwise, the MCP intrinsic DMPALL program is scheduled to generate the routine described in the <specification string>.

<specification string>

The available options and their methods for inclusion in the <specification string> are defined under Utility Programs. See the DMPALL intrinsic in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

When executing DMPALL, the semicolon delimiter must be used following the <specification string> when <additional control instructions> such as MEMORY or PRIORITY are desired.

Examples:

GENERATE /OWN CRDCRD A B; AFTER PROG1; CHARGE 99

executes a user-provided version of DMPALL to generate a program to copy a card file named A to a punch file named B. This function will be performed after PROG1 finishes executing. A charge number of 99 will be inserted into the MCP Run log for these programs.

GEN LIST TAPFIL 200 100; MEM 45

executes the MCP version of DMPALL to generate a program that will provide a double-spaced listing of the tape file named TAPFIL. This MEM command specifies that this task will be allocated 45 KD of memory to execute.

See also:

PERFORM (Schedule DMPALL) command.

The DMPALL utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

GO (Restart A Stopped Program)

Keyboard Input Message

Access Level: 3

The GO command requests the resumption of a program stopped by an ST (Suspend Program Processing) or STOP (Suspend Program Processing) command.

The GO command has the same effect as the START (Restart A Stopped Program) command.

— <mix no> GO —|

Figure 2-97. GO Command Syntax

<mix no>

is the mix number of the program to be restarted.

Details

The program restarts its execution at the point where it stopped. It starts with the same values it was using when it stopped.

If you issue GO on a program that is not stopped, the system ignores the command.

If There Is Not Enough Memory

If there is not enough memory for the program being restarted:

- Memory is allocated by priority. If you raise the program's memory priority, it might restart.
- When sufficient memory becomes available, the program will restart.
- A program being restarted will start before a program of the same priority that is waiting in the schedule.
- Programs with lower memory priority are rolled out from memory to disk if this will make enough room for the program that is waiting to restart.

Example

119 GO

restarts a program with mix number 119.

See also:

ST (Suspend Program Processing); START (Restart A Stopped Program); and STOP (Suspend Program Processing) commands.

GT (Initiate Trace)

Keyboard Input Message

Access Level: 3

The GT command initiates the instruction trace for a specified program that is executing.

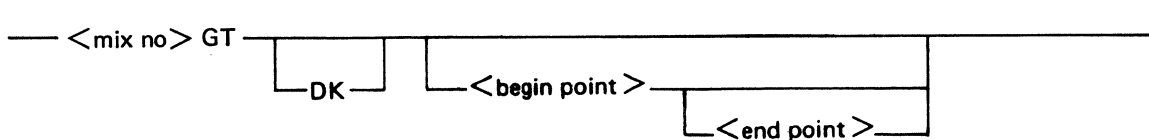


Figure 2-98. GT Command Syntax

<mix no>

designates the mix number of the program to be traced and cannot be 0 or 1.

DK

causes a printer backup disk file to be used for the trace data.

DK has the same effect as using the OU (Direct File To Device) command to send the file to disk or as setting the GTDK system option. The GTDK option can be set through the USE GTDK record in the system configuration file or with an SO (Set System Options) command.

If this option is omitted, the trace goes to the system default medium, determined by the setting of the GTDK system option.

<begin point>

<end point>

allow the operator to specify beginning and ending limits for the trace printing.

If the <begin point> is specified, the trace begins printing when the program reaches the designated point.

If the <end point> is also specified, the printing terminates at that point and does not resume until the <begin point> is again reached. The program is still being traced during this period, but the printing is suppressed.

If you omit <begin point> and <end point>, the entire program can be traced. Tracing will begin where the program was executing when the GT was entered. This will trace user code only.

The format for the <begin point> and <end point> specifications is shown in Figure 2-99.



Figure 2-99. Begin/End-Point Specification Syntax

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

<segment no> /

<instruction address>

must be the exact segment number and address of an instruction. If either is 0, the corresponding limit is not enabled. A <segment no> of 0 means any segment, not just segment 0.

Details

The trace is listed on the first available line printer; if none is available, any printer in use by the designated program is used. If printers are not available, the program is suspended until a printer becomes available or until the operator directs the trace to backup disk through an OU (Direct File to Device) command.

You can use the TRACE (Initiate Trace) command to trace a program when it enters the mix.

The NT (Terminate Trace) command stops the tracing of a program.

Traces can also be started programmatically. The TRACE program branch communicate (BCT) requests that the MCP initiate or terminate a program trace or to produce a memory dump of the program. See the *V Series Program Interfaces Programming Reference Manual* for more information.

The interactive DEBUG module has a number of sophisticated debugging features that can be used to debug user programs. DEBUG (Attach Program To A DEBUG Session) and ID (Invoke DEBUG) can be used to start DEBUG sessions. GT and TRACE can be used only for user programs.

Examples

21 GT DK

starts an instruction trace on the program with mix number 021 and sends the trace data to a printer backup file on disk.

14 GT 13/1484 13/14620

starts an instruction trace on the program with mix number 014. The trace should begin at segment 13, address 1484 and end at segment 13, address 14620.

See also:

DEBUG (Debug Session); ID (Invoke DEBUG); NT (Terminate Trace); OU (Direct File To Device); SO (Set System Options); and TRACE (Trace A Program) commands.

The USE GTDK record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *V Series Program Interfaces Programming Reference Manual* and the various program language manuals for information about TRACE.

HL (Request A Halt/Load)

Keyboard Input Message

Access Level: 8

The HL command causes a halt/load or halt operation to be performed. You can use HL only at the HLSPO ODT.

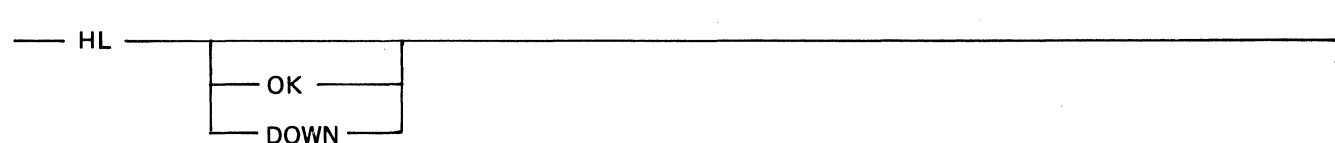


Figure 2-100. HL Command Syntax

HL
causes a halt/load operation. HL is valid only if there are no jobs running in the mix.

OK
must be entered if jobs are running in the mix and you want to perform a halt/load operation.

CAUTION

Be aware that HL OK will terminate all jobs and related activities on the system.

DOWN
causes just the halt portion of a halt/load operation. HL DOWN is valid only if there are no jobs running in the mix.

Details

A halt/load operation can do the following:

- load the MCP from the system disk into memory
- build the tables and lists in memory used by the operating system

A halt/load is usually used to start the machine and to recover from failures.

The HLSPO is an Operator Display Terminal (ODT) that displays halt/load messages and fail codes in the event of system failure. The HLSPO is declared in the UNIT ODT record of the system configuration file or through the UNIT (Add Peripheral Devices) command.

If you enter HL or HL DOWN and there are jobs running in the mix, the message stating that a null mix is required is displayed.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

HL

causes a halt/load operation.

HL OK

causes a halt/load operation even though jobs are running in the mix.

HL DOWN

causes the halt portion of a halt/load operation.

See also:

UNIT (Add Peripheral Devices) command.

Halt/loads and the UNIT ODT record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

HN (Identify Or Change Hostname)

Keyboard Input Message

Access Level: 0

The HN command can be used identify and change the hostname of the local processor that supports BNA network architecture.

The hostname is originally supplied during system initialization through the HOSTNAME record.

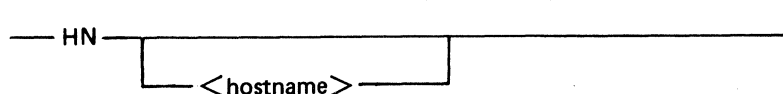


Figure 2-101. HN Command Syntax

HN

can be used without the <hostname> option to display the current host name.

<hostname>

identifies the host and can be up to 17 characters long as follows:

- a number followed by up to 16 additional numbers.
- a letter followed by up to 16 characters, which can be any combination of numbers, uppercase letters and lowercase letters.

You cannot change the hostname if users are running. You must have a null mix. However, you can display the name of the current host at any time.

Details

The host name is used to identify the processor in the Run Log, the ODT log, and the Maintenance log.

Examples

The following examples contain a single line as entered at the ODT followed by the one line response from the system. The original host name is "RED".

HN	HN = RED
HN GREEN	HOSTNAME CHANGED FROM RED TO GREEN
HN 123	HOSTNAME CHANGED FROM GREEN TO 123
HN BNAh1	HOSTNAME CHANGED FROM 123 TO BNAh1

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Invalid examples:

HN 1HOST

is invalid because you cannot have a host name that starts with a number and is followed by letters. An invalid syntax message will be displayed.

HN HOST-1

is invalid because you cannot have a special character such as a hyphen as part of a host name. An invalid syntax message will be displayed.

See also:

The HOSTNAME record in *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *V Series BNA Installation and Operations Guide*.

The *V Series MCP/VS Programming Reference Manual*.

ID (Invoke DEBUG Session)

Keyboard Input Message

Access Level: 8

The ID command invokes a DEBUG session at the ODT on which it is entered.

— <mix no> ID —

Figure 2-102. ID Command Syntax

<mix no>

refers to the mix number of an active job in the mix.

Related Commands

The ID command, which invokes a DEBUG session, differs from the DEBUG command. DEBUG starts a program and attaches that program to an interactive DEBUG session.

The ED and QD commands, like the ID command, manipulate DEBUG sessions. The ED (Enter Debug Session) command allows you go back into a DEBUG session. The QD (Query Debug Sessions) command displays the number of active DEBUG sessions on the system with their corresponding session numbers.

Example

55 ID

invokes DEBUG for the program associated with mix number 055.

See also:

DEBUG (Attach A Program To A DEBUG Session); ED (Enter DEBUG Session); and QD (Query DEBUG Sessions) commands.

The DEBUG utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

C

2-155

ON <pack name>
indicates that the file resides on a particular pack.

<deck no>

<cc/u>

can be specified if the file to be assigned is present in an active pseudo reader. <cc/u> indicates the required pack.

<unit id>

is a one- to six-character name that can be used for data communications on MICR files to specify the desired device. The file-id must match the UNIT-ID declared on a UNIT record in the system configuration file.

For disk or pack files, use the IL command syntax shown in Figure 2-104. Note that this syntax can be used only in response to a file-not-in-directory or duplicate-file-on-pack message. This syntax can also be used for pack files if the pack family name is correct but a different file on the family is to be used.

— <mix no> IL <file identifier> —

Figure 2-104. IL Command Syntax (Disk, Pack Files)

<mix no>

is the mix number of the program requesting the file.

<file identifier>

is the name contained in the disk or pack directory. File-identifiers can be up to six characters long. They cannot be made up of all nulls, all blanks, or all zeros.

Details

If you know at run time that the file needed by the program has a different name or location than the program expects, the FILE (Equate File-Name with File-Identifier Or Medium) command could be used to redirect the program.

Examples

4 IL 3/0

gives the program associated with mix number 004 the location of the labeled file it needs on channel number 3, unit number 0.

21 IL SYMFIL

gives the program associated with mix number 021 the name of the labeled file it needs as SYMFIL.

32 IL FILE01 ON DMSPAK

gives the program associated with mix number 032 the name of the labeled file it needs as FILE01, which resides on the pack named DMSPAK.

See also:

FR (Final Reel); OF (Indicate Optional File); and UL (Assign Unlabeled File) commands.

IN (Insert Data Into Program)

Keyboard Input Message

Access Level: 3

The IN command loads a data string into a job in the mix at the indicated address.

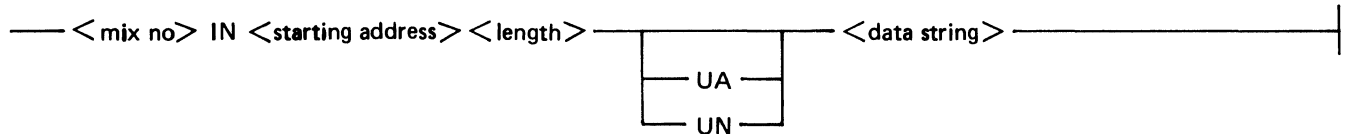


Figure 2-105. IN Command Syntax

<mix no>

is the mix number of the program into which you want to insert the data string.

Note that you cannot specify a locked program or use a mix number of 0 or 1.

<length>

indicates the number of digits (for UN data) or bytes (for UA data) to be inserted and must be 30 or less.

<starting address>

is the program base-relative address at which to start inserting. This can range from 000000 through 999999.

UN

UA

indicate data class.

UN specifies unsigned numeric data and causes digits and undigits to be inserted. Undigits are the hexadecimal values between 10 and 15. The undigits are A, representing hexadecimal 10, through F, representing hexadecimal 15.

UA denotes unsigned alphanumeric data. This inserts characters with the <length> in bytes.

If a data classification is not specified, UN is assumed unless the data is contained in quotes. UA is optional if the data is enclosed in quotation marks.

<data string>

is the information to be inserted into the program.

If you use special characters (period, comma, and so forth), the data is considered to be in UA format and must be enclosed in quotation marks

Two consecutive quotation marks within the <data string> cause a single quotation mark to be inserted.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Details

The IN command can be used on a program that is already executing, which differs from the INSERT (Insert Data Into Program) command, which is used prior to a program's execution.

Examples:

10 IN 16 2 56

inserts 2 digits of UN or numeric data with a value of 56 into the program with mix number 010 beginning at address 16.

7 IN 0 8 UA ABCDEF12

inserts 8 bytes of alphanumeric data with the value of ABCDEF12 into the program with mix number 007 beginning at address 0.

7 IN 0 8 "ABCDEF12"

inserts 8 bytes of alphanumeric data with the value of ABCDEF12 into the program with mix number 007 beginning at address 0.

See also:

INSERT (Insert Data Into Program); and SW (Set Programmatic Switches); VALUE (Insert Value into a Program) commands.

INSERT (Insert Data Into Program)

MCP Control Instruction

Access Level: 1

INSERT places required data into a program prior to its execution.

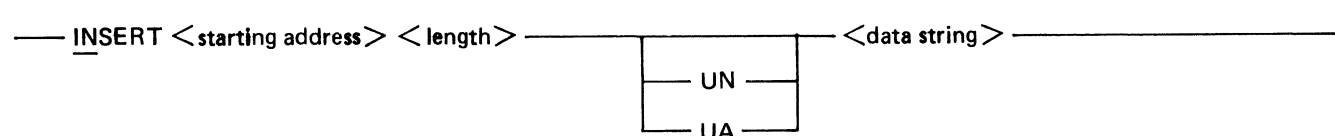


Figure 2-106. INSERT Command Syntax

INSERT

IN

are synonymous.

<starting address>

is the program base-relative address at which to start inserting. This can range from 000000 through 999999.

NOTE

In most cases, the <data string> inserted into a program prior to its execution must not terminate at an address greater than 000063. Data inserted into higher base-relative addresses is destroyed by the overlay mechanism generated by most compilers.

<length>

indicates the number of digits (UN) or bytes (UA) of data to be inserted and must be 30 or less.

UN

UA

indicate data class.

UN specifies unsigned numeric data and causes digits and undigits to be inserted. Undigits are the hexadecimal values between 10 and 15. The undigits are A, representing hexadecimal 10, through F, representing hexadecimal 15.

UA denotes unsigned alphanumeric data. This inserts characters with the <length> in bytes.

If a data classification is not specified, UN is assumed unless the data is contained in quotes. UA is optional if the data is enclosed in quotation marks.

<data string>

is the information to be inserted into the program.

If you use special characters (period, comma, and so forth), the data is considered to be in UA format and must be enclosed in quotation marks

Two consecutive quotation marks within the <data string> cause a single quotation mark to be inserted.

Details

IN can be used on a program that is already executing, which differs from the INSERT (Insert Data Before Execution) command, which is used at program initiation time.

Examples:

EXECUTE PROG1; INSERT 0 8 UN 12A00011

executes PROG1 and inserts 8 digits of unsigned numeric data with the value of 12A00011 beginning at address 0.

EXECUTE PROG2; INSERT 10 4 UA ABCD

executes PROG2 and inserts 4 bytes of unsigned alphanumeric data with the value of ABCD beginning at address 10.

EXECUTE PROG3; INSERT 0 3 UA "A""B"

executes PROG3 and inserts 3 bytes of unsigned alphanumeric data with the value A"B beginning at address 0.

See also:

IN (Insert Data Into Program); SW (Set Programmatic Switches); and VALUE (Insert Value Into A Program) commands.

IR (INITIATE DMSII RECOVERY)

Keyboard Input Message

Access Level: 8

The IR command initiates certain database functions. These include database recovery, reorganization of database files, execution of Production DBPs, purging of database structures, and the loading and dumping of database archive tapes.

For a complete description of the syntax and usage of this command, see the V Series DMSII Operations Reference Manual Volume 2: Administration.

KA (ANALYZE DISK DIRECTORY)

Keyboard Input Message

Access Level: 5

The KA function invokes the intrinsic program DSKOUT to generate an analysis of all files found in the disk directory. The following information appears for each file:

- the logical ID
- area address
- file identifier
- area numbers, total areas used, and maximum areas declared
- total number of segments used
- logical record size
- end of file pointer
- file attributes

You can specify a single file id , a group file id, or a subsystem number. The system will analyze only the relevant file(s).

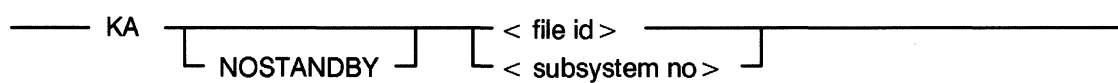


Figure 2-107. KA Command Syntax

■ NOSTANDBY

directs the operating system to capture information for DSKOUT even when the operating system *cannot* perform its normal standby function. The standby function is still attempted, but information is captured regardless of whether the function is successful. For more information about the standby function, see Details on the following pages.

<file id>

refers to the file id or masked file id of files on which an analysis will be performed. The files can reside on any of the existing disk subsystems on the system.

<subsystem no>

analyzes only the files with areas on that disk subsystem and can be a number from 1 to 8.

Output Options

By default, output for the KA and PA commands is sorted by file id, and output for the KS (Analyze Disk Space) and PS commands is sorted by address. There are several options, however, for specifying the output. DSKOUT can provide the following:

- Output sorted by total file size
- Output sorted by file id (alphabetically)
- Output sorted by address (default)
- Output sorted by available area size
- Calculation of wasted space in files, sorted by total file size
- Identification and analysis of code files
- Provision of installation name on printout

These options are specified at execution time with the VALUE statement:

VALUE 1 <integer>

The following list of options will appear on the ODT if <integer> = 1:

DSKOUT= <MX>	*** ENTER OPTIONS ***
FILESIZE	SORT BY FILESIZE (KA, PA ONLY)
NAME	SORT BY NAME (KA, PA ONLY)
ADDRESS	SORT BY ADDRESS (KA, KS, PA, PS ONLY)
AREASIZE	SORT BY AVAILABLE AREA SIZE (KS, PS ONLY)
WASTED	ANALYZE WASTED SPACE (KA, PA ONLY)
CODE	ANALYZE CODE FILES (KA, PA ONLY)
INSTALL	ACCEPT INSTALLATION NAME (KA, PA ONLY)
** DSKOUT=	<MX> ACCEPT

If <integer> = 8, the options will not be displayed on the ODT, but DSKOUT will still wait for an ACCEPT.

The response to the accept should be any valid option or combination of options, in string format. Valid option delimiters are a comma, slash, or space. For example:

```
<MX> AX "NAME, ADDRESS"
<MX> AX "FILESIZE ADDRESS CODE"
<MX> AX "AREASIZE/INSTALL"
```

If invalid input is received, DSKOUT will display an error message on the ODT, redisplay the valid options, and then wait for another ACCEPT. This will continue until you enter valid options or use the DS (Discontinue Program) command to terminate the program.

The following restrictions apply to options:

<i>OPTION valid only with these</i>	<i>COMMANDS</i>
FILESIZE	KA, PA
NAME	KA, PA
ADDRESS	KA, KS, PA, PS
AREASIZE	KS, PS
WASTED	KA, PA
CODE	KA, PA
INSTALL	KA, KS, KP, PA, PS, PP

NOTE

The CODE option is VERY time consuming, because each file is opened to have the program parameter block and all file parameter blocks read.

The default option is ADDRESS for KS/PS and NAME for KA/PA.

■ Details

■ When you enter a KA command, the operating system can respond with the message:

■ KBD IGNORED: DISK OR PACK MAINT IS IN PROGRESS

■ This means the operating system cannot perform a standby function as part of the KA command. A standby function suspends updates to the disk directory and other data structures long enough to obtain an accurate “snapshot” of the structures. This “snapshot” is passed to DSKOUT for analysis. The operating system needs exclusive access to the disk directory for an accurate DSKOUT analysis.

■ To respond to this situation, you can:

- • Wait until a later time when disk activity is less
- • Use the NOSTANDBY option of the KA command

■ NOSTANDBY captures information for DSKOUT even when the operating system *cannot* perform the standby function. The standby function is attempted, but information is captured regardless of whether it is successful.

■ Using NOSTANDBY can create missing or duplicate disk messages that are not correct. This happens if an update occurs as the “snapshot” is taken. If NOSTANDBY is used and missing/duplicate disk conditions are encountered, DSKOUT flags the resulting messages with NOSTANDBY warnings.

■ These warnings mean the condition may be caused by non-exclusive access rather than actual missing or duplicate disk. Before you act on any missing/duplicate disk messages with a NOSTANDBY warning, enter another KA command without NOSTANDBY. Enter the command when less disk activity is taking place.

Examples

KA =A=BCD

causes the system to analyze the files that fit the masked file id of =A=BCD.

KA HFILE

causes the system to analyze HFILE.

KA 5

causes the system to analyze the files with areas on disk subsystem number 5.

KA NOSTANDBY 7

causes the system to analyze the files with areas on disk subsystem number 7. The analysis is generated even if the normal standby function of the KA command is unsuccessful.

See also:

KP (Print Disk Segments), KS (Analyze Disk Space), PA (Analyze Pack Directory), PP (Print Pack Sectors), and PS (Analyze Pack Space) commands.



KP (PRINT DISK SEGMENTS)

Keyboard Input Message

Access Level: 5

KP invokes the intrinsic program DSKOUT to print selected segments from a disk subsystem. The disk ID, starting segment address, and number of segments specify which segments to print.

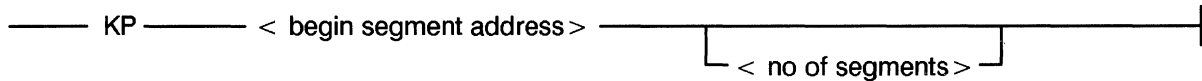


Figure 2-108. KP Command Syntax

<disk ID>

specifies a previously declared disk unit and can be a number from 1 to 255.

<begin segment address>

specifies the beginning point for the printing, must be stated as an integer, and must be 0 to drive capacity or the declared limit.

<no of segments>

indicates the number of segments you want to print. If you omit this option, the system assumes a value of one.

Examples

KP 3 10

invokes the DSKOUT intrinsic to print 1 disk segment beginning with segment 10 from disk ID 3.

KP 2 1557 25

invokes the DSKOUT intrinsic to print 25 disk segments beginning with segment 1557 from disk ID 2.

KP 1 1489 50; PR 5

invokes the DSKOUT intrinsic to print 50 disk segments beginning with segment 1489 from disk ID 1 and assign it a memory, schedule, I/O and processing priority setting of 5.

See also:

KA (Analyze Disk Directory), KS (Analyze Disk Space), PA (Analyze Pack Directory), PP (Print Pack Sectors), and PS (Analyze Pack Space) commands.

KS (ANALYZE DISK SPACE)

Keyboard Input Message

Access Level: 5

KS invokes the intrinsic DSKOUT to analyze the allocation of disk space. Files, available areas, missing and duplicate disk are listed. The following are listed for each file:

- logical ID
- area addresses
- file identifier
- area number and size of area
- file attributes

For available areas, the logical ID, area address, the string "A V A I L A B L E", and the size of the available space are listed. For missing or duplicate disk, the logical ID, address of the area, the string "MISSING DISK" or "DUPLICATE DISK", and size of the area are listed.

You can generate an analysis of only a specified subsystem or analysis of only available space on disk.

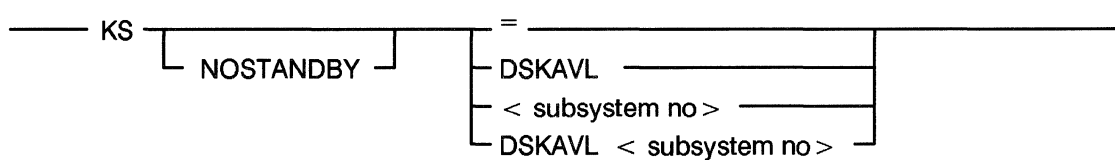


Figure 2-109. KS Command Syntax

■ NOSTANDBY

directs the operating system to capture information for DSKOUT even when the operating system *cannot* perform its normal standby function. The standby function is still attempted, but information is captured regardless of whether the function is successful. For more information about the standby function, see Details on the following pages.

=

causes an analysis of all disks on the system.

DSKAVL

causes an analysis of only the unassigned disk space on the system or on the specified subsystem.

<subsystem no>

causes an analysis of only the specified disk subsystem.

Output Options

Various output options to provide alternate sorting and analysis are available. See the KA (Analyze Disk Directory) for a full description of these options and their applications.

Details

When you enter a KS command, the operating system can respond with the message:

KBD IGNORED: DISK OR PACK MAINT IS IN PROGRESS

This means the operating system cannot perform a standby function as part of the KS command. A standby function suspends updates to operating system data structures long enough to obtain an accurate "snapshot" of them. This "snapshot" is passed to DSKOUT for analysis. The operating system needs exclusive access to the data structures for an accurate DSKOUT analysis.

To respond to this situation, you can:

- Wait until a later time when disk activity is less
- Use the NOSTANDBY option of the KS command

NOSTANDBY captures information for DSKOUT even when the operating system *cannot* perform the standby function. The standby function is attempted, but information is captured regardless of whether it is successful.

Using NOSTANDBY can create missing or duplicate disk messages that are not correct. This happens if an update occurs as the "snapshot" is taken. If NOSTANDBY is used and missing/duplicate disk conditions are encountered, DSKOUT flags the resulting messages with NOSTANDBY warnings.

These warnings mean the condition may be caused by non-exclusive access rather than actual missing or duplicate disk. Before you act on any missing/duplicate disk messages with a NOSTANDBY warning, enter another KS command without NOSTANDBY. Enter the command when less disk activity is taking place.

Examples

KS =

invokes the DSKOUT intrinsic to analyze disk space on all the disks on the system.

KS DSKAVL; PR = 9; LOCK

invokes the DSKOUT intrinsic to analyze only the unassigned disk space on the system; assigns the job a memory, schedule, I/O, and processing priority of 9; and locks the job, thus preventing it from being discontinued or modified.

KS NOSTANDBY 6

invokes the DSKOUT intrinsic to analyze disk space on disk subsystem 6. The analysis is generated even if the normal standby function of the KS command is unsuccessful.

See also:

KA (Analyze Disk Directory), KP (Print Disk Segments), PA (Analyze Pack Directory), PP (Print Pack Sectors), and PS (Analyze Pack Space) commands.

KX (OBTAIN USER DISK SPACE)

Keyboard Input Message

Access Level: 0

The KX command removes disk files to make disk space available. This is generally done in response to a no-user-disk message.

The KX command lists the files that can be removed. The files are ordered as they appear in the directory. For each file found, a line indicates the name of the file, segments per area, maximum number of contiguous disk segments, and the subsystem on which the file resides. If the SECURITY option is set, the usercode of each file is also displayed. You can then keep or remove the file.

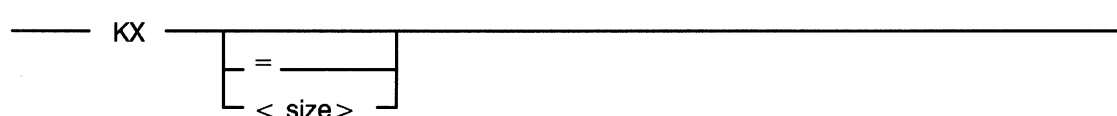


Figure 2-110. KX Command Syntax

KX
KX =

are synonymous and display a list of removable files in the directory. Removable files are those not in use and whose security attributes allow library maintenance and LIBMAINT is on. See the *V Series System Software Security Installation and Operations Guide* for more information.

<size>

specifies that only files containing at least the indicated number of contiguous disk segments are to be listed.

A KX command without the <size> option has zero as the number of contiguous segments and hence will list all files.

Responses to KX

There are four possible responses to each line of output from the KX request:

- A null response (ETX) leaves the listed file on disk and continues searching for another file to remove.
- RM removes the listed file from disk. However, the file might be opened between the time it is listed and the time the RM is entered and, therefore, it cannot be removed. If the file was successfully removed, the MCP continues to search for files. If the file has not been removed, a message is displayed saying that the file is not removed, and MCP continues to search for files.
- KX <size> restarts the KX with a new size. The listed file remains on disk. Although <size> is required, it may be zero.
- END terminates the KX attempt. The listed file remains on disk.

AUTO KX

If you have run out of disk space and the MCP tries to add more data to the disk, you can trigger an AUTO KX. This acts similarly to the KX command in that you have to analyze the disk directory and remove files to obtain the needed disk space.

If a response of END is given to an AUTO KX and insufficient disk space has been made available, the AUTO KX is reinitiated.

CAUTION

Do not perform a halt/load while an AUTO KX is in progress. If AUTO KX is invoked due to an MCP log transfer and a halt/load is performed to terminate the KX, it may corrupt disk directories and data files.

Examples

KX =

lists all removable files. This can be viewed as a KX specifying zero as the number of contiguous segments.

KX 4000

lists only files with at least 4000 contiguous disk segments.

See Also:

KS (Analyze Disk Space); RX (Return Deleted Space); and WX (Display Removed Space) commands.

LABEL1 (Change Tape Label Format)

MCP Control Instruction

Access Level: 0

The LABEL1 command changes the format of an installation-defined magnetic tape label.

The LABEL1 instruction must be entered through a card reader, and the punched card must contain an invalid character in column 1. The word LABEL1 can appear anywhere in columns 2 through 8.

—— LABEL1 <label parameters> ——|

Figure 2-111. LABEL1 Command Syntax

<label parameters>

are the parameters you can enter to customize the tape label.

The record is 67 columns long.

Note that if multifile-identifiers are specified, the length of the file-identifier must be six characters or less. However, if the length of the MFID is 0, the file-identifier can be up to 12 characters long. The 10, 11, and 12 character long file-identifiers are specified by the letters A, B, and C, respectively.

V Series MCP/VS System Software Operations Guide, Volume 2
The System Commands

COLS	CONTENTS	NOTES
1-8	LABEL1	The control word LABEL1 can appear anywhere in the field
9	Blank	Skip this column
10-19	1-10 char	Text of beginning of file label identifier
20-21	0-99	Position of beginning of file label identifier field
22-23	1-10	Length of beginning of file label identifier field
24-25	0-99	Position of multifile id field
26	0-6	Length of multifile id field (0 = none)
27-28	0-99	Position of file id field
29	1-C (C=12)	Length of file id field
30-31	0-99	Position of reel number field
32	1-9	Length of reel number field
33-34	0-99	Position of creation date field
35-36	0-99	Position of purge date field
37	3-9	Length of purge date field (a value of less than five implies a retention period instead of purge date)
38-39	0-99	Position of volume id field
40	1-9	Length of physical volume id field
41-42	0-99	Position of end of reel or end of file identifier
43	1-5	Length of end of reel or end of file label identifier
44-48	1-5 char	Text of end of reel label id
49-53	1-5 char	Text of end of file label id
54-56	20-998	Minimum length of label records

V Series MCP/VS System Software Operations Guide, Volume 2
The System Commands

COLS	CONTENTS	NOTES
57-58	0-99	Number of label records (0=variable)
59	0-1	1=tape mark after end labels
60	0-1	1=tape mark after beginning labels
61-62	0-99	Position of block count field
63	0-9	Length of block count field (0=none)
64-65	0-99	Position of record count field (end labels only)
66-67	0-12 (0=none)	Length of record count field

Example

LABEL1 **BEGIN**001018659642368545114745*ENDR*ENDF20001114653107

changes the format of an installation-defined tape label.

Columns 1-8 contain LABEL1; 9 is blank.

Columns 10-19 contain **BEGIN** as the beginning of file label identifier. Columns 20-21 have 00 as the position of the beginning of the file label identifier. Columns 22-23 have 10 as the length of the beginning of file label identifier.

Columns 24-25 contain 18 as the position of the multifile id; column 26 has 6 as the length of the multifile id.

Columns 27-28 has 59 as the position of the file id; column 29 has 6 as the length of the file id.

Columns 30-31 have 42 as the position of the reel number; column 32 has 3 as its length.

Columns 33-34 have 68 as the position of the creation date; columns 35-36 have 54 as the position of the purge date; and column 37 has 5 as the length of the purge date.

Columns 38-39 have 11 as the position of the volume id; column 40 has 4 as its length.

Columns 41-42 have 74 as the position of the EOR or EOF identifier; column 43 its length. Columns 44-48 contain *ENDR as the text of the EOR label identifier; columns 49-53 have *ENDF as the text of the EOF label identifier.

Columns 54-56 have 200 as the minimum length of label records; columns 57-58 have 01 as the number of label records.

Columns 59 and 60 each contain a 1 to specify a tape mark after end labels and after beginning labels, respectively.

Columns 61-62 contain 46 for the position of the block count; column 63 has 5 as the length of the block count.

Columns 64-65 have 31 as the position of the record count; columns 66-67 have 07 as the length of the record count.

See Also:

AC (Purge And Place Tape Number In Label); CN (Display Tape Number); SN (Purge And Place Tape Number In Label); and TM (Write Tape Mark) commands.

The INSTALLATION LABEL1 record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

LC (INSERT LOG COMMENT)

Keyboard Input Message

Access Level: 1

- The LC command inserts a comment into a system log file.

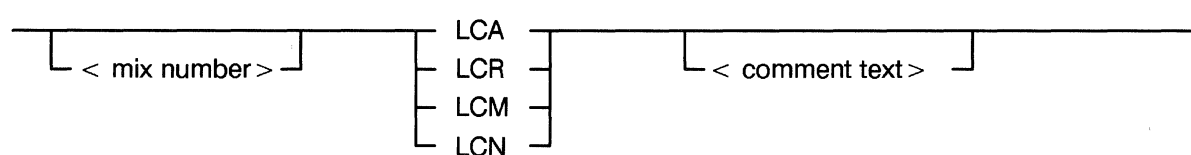


Figure 2-112. LC Command Syntax

- <mix number>

is the mix number of a currently active, waiting, or stopped job. You cannot use a mix number of 0 or 1, which indicates the MCP. The <mix no> can be omitted if you do not want to associate the comment with a particular job.

- LCA

inserts the comment in all logs.

- LCR

inserts the comment in the Run Log and the ODT log.

- LCM

inserts the comment in the Maintenance Log and the ODT log.

- LCN

inserts the comment in the Network Log.

<comment text>

contains the text of the comment and can be up to 75 characters long. The <comment text> must be enclosed in quotes if it contains non-alphanumeric characters.

Details

To use this command, the requested log must be in use. See the USE LOGGING records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The comment is placed in a record type 5/0 in the Run Log and the Maintenance Log, if specified.

Examples

LCA START OF SHIFT

places this comment in all logs. Note that the comment does not have to be enclosed in quotes because it contains only alphanumeric characters.

12 LCR "GIVEN WRONG FILE - DSED"

places this comment in the Run Log. The quotes are required because the comment contains a hyphen. This comment is associated with the job with mix number 012.

LCN STATION X2594 DISABLED FOR RELOCATION

places this comment in the Network Log. Quotes are not required.

LCM "2/3 TAKEN OUT OF USE - PARITY ERRORS"

places this comment in the Maintenance Log. The quotes are required because the comment contains a slash.

See also:

The USE RLOG Record, USE SLOG Record, USE MLOG Record, and USE NLOG Record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *V Series System Software Logging Operations Reference Manual*.

LD, LDP (CREATE PSEUDO CARD DECK)

Keyboard Input Message

Access Level: 7

The LD command schedules the LDCNTL intrinsic program, which reads card files and converts them into pseudo card files on disk or pack.

The LDCNTL program can also be scheduled through a CV (Convert Punch Backup File To Pseudo Card Deck) or an EXECUTE (Execute Library Program) command.

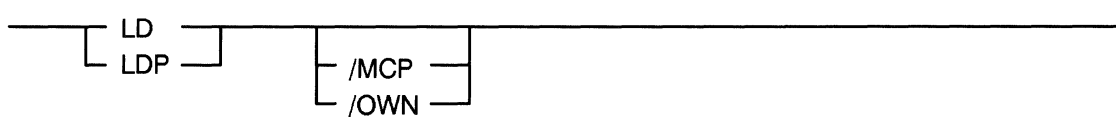


Figure 2-113. LD, LDP Command Syntax

LD
LDP

LD directs the pseudo deck to disk; LDP directs them to pack.

/MCP
/OWN

/MCP executes the version bound to the MCP and is the default. /OWN executes a free-standing user-written version.

If this option is omitted, the user-coded version is initiated if present on disk, otherwise the MCP intrinsic is initiated.

Examples

LD; AFTER MYPROG; PRIORITY 5

starts LDCNTL after a program called MYPROG finishes and assigns it a memory, processing and I/O, and schedule priority of 5.

See Also:

CV, CVP (Convert Punch Backup File To Pseudo Card Deck); DA (Deactivate Pseudo Reader); EXECUTE (Execute Library Program); OL (Display Peripheral Status); and RN, RNP (Activate Pseudo Reader) commands.

The LDCNTL utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

LH (LOAD FIRMWARE TO CHANNEL CONTROLLER / DCP)

Keyboard Input Message

Access Level: 8

The LH command invokes the utility program LDHOST to download firmware to a programmable controller, a Data Communication Processor (DCP), or a UNILINE device. In the case of a B974 DCP, the LH command invokes the utility program B974LD. In the case of communications devices connected through the V Series Communication System (VCS), LH invokes the utility program CPLOAD.

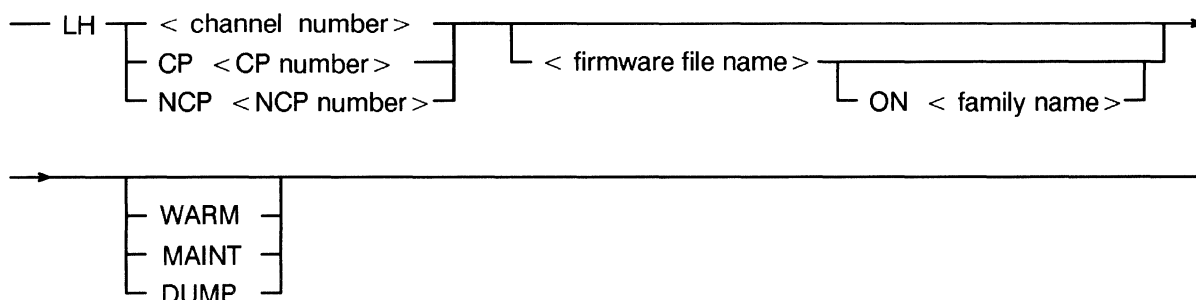


Figure 2-114. LH Command Syntax

<channel no>

specifies the channel to which the controller is connected. It can be a secondary channel on an exchange. This option is not used if the device on the channel is connected through the V Series Communication System (VCS).

CP <CP number>

loads firmware to a Communication Processor (CP) connected through VCS. Examples of CPs that can be connected through VCS include a CP 3680 Front End Processor or a TELCOM DLP. This option is used for data communications devices that have a physical connection to individual stations.

NCP <NCP number>

loads firmware to the communications processor (CP) connected to the entered Network Communications Port (NCP) that is connected through VCS. This option functions the same as the CP <CP number> option.

<firmware file name>

is the name of the firmware file to be loaded. The firmware file must be on disk when the LH command is issued.

This option is not required if standard firmware or a DCP, as specified in the NDL, is being host loaded to the controller. Standard firmware includes that specified on the DLP record of the system configuration file.

For the CP <CP number> or NCP <NCP number> options, this is the name of the VCS initialization file.

■ ON <family name>

This option is valid only when used with the NCP <NCP number> or CP <CP number> options of the LH command. The option indicates the diskpack family where the firmware file for the NCP or CP resides.

■ WARM

can be used only with DCPs. It specifies that the internal DCP tables are not to be reloaded. This can allow recovery from an error condition without restarting the entire data comm network.

■ MAINT

can be used with any hardware type to specify that the channel is to be left inhibited at the completion of the load. It allows the loading of diagnostic firmware (for example, a pack exerciser) to one controller on an exchange while continuing normal operations on the other exchange channels.

■ DUMP

This option is valid only when used with the NCP <NCP number> or CP <CP number> options of the LH command. This option causes the memory of the data communications device to be stored in a file before firmware is loaded. The file resides on the V Series system. The default value does not request a memory dump.

NOTE

WARM and MAINT must be used in separate commands. They cannot be used together. The following table shows host load files and their usage.

Table 2-10. Host Load Files And Their Usage

<i>Name</i>	<i>Usage</i>
HSTLAD	Loads firmware to B9389 controller to 680 or 682 pack via SEQ-HT
HSTLKC	Loads firmware to B 9384, B 9385 controller to 225 pack via HT-DLP
HSTLQH	Loads firmware to B 9387 controller to 206/207/677/659 pack via HT-DLP
HSTLUD	Loads firmware to B 9387 controller to 206/207/677/659 pack via SEQ-HT. If controller is not correctly updated for the HST-LUD version, firmware is "changed" to HSTLUC version.
USP2BH	Loads UNILINE SPO firmware
UST2BH	Loads UNILINE terminal firmware
IPPFMW	Loads Loadable Operating System (LOS) firmware to an IPP
LSDIAG	Load Loadable Diagnostics to an IPP

MD4 and MD8 disk/pack does not utilize a controller and, therefore, does not require any controlware to be loaded.

Uses

A DCP can be host loaded when the DCP option is reset, the DCP is marked "LH REQUIRED", or the DCP is not in use.

If a B974 or B874 is not working properly because its data communication code has become corrupted, the MCS programs will receive indication that the DCP is not responding. The LH command will reload the code into the B974 or B874.

You might want to use the LH command if the NDL source has been recompiled and is to be loaded into one particular B974. This could be desirable if different versions of the NDL are being used on different DCPs.

The CP <CP number> and NCP <NCP number> options cannot be used unless VCS is running on the system.

Details For The B974

If the LH command initializes a B974 DCP, it automatically dumps its S-memory to the host's disk with the file name DCccsP. In this case, cc is the channel number of the DCP, and the s is the system number of the host. The dump, if one exists, uploads to the host with the file name DCdnnP. In this case, d is the DCP number of the B974, and nn is the B974's dump number (Range 00-05).

Example

LH 2 HSTLQH

invokes LDHOST to load the programmable controller on channel 2.

See also:

LDHOST in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.



LI (LOG IN)

Keyboard Input Message and MCP Control Instruction

Access Level: 0

The LI command logs an ODT or card reader into the system. This command has to be entered from the device that is being logged in.

The syntax as shown in Figure 2-115 is the same for either LI command.

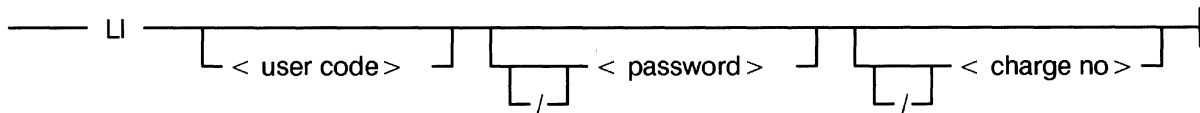


Figure 2-115. LI Command Syntax

<usercode>

refers to a code that identifies a user to the system.

<password>

refers a group of characters by which the system determines the legality of a user.

<charge no>

refers to a number used to assign charges for system use.

NOTE

The <usercode>, <password>, and <charge number> are all optional. Slashes are required only if leading parameters are omitted (for example, LI //90023).

System Security

System Security refers to system access control and disk file security functions. System access control restricts access to the system and defines each user's capabilities on the system. Disk file security controls programmatic and operator access to permanent disk and pack files. In addition, disk file security allows different file access mode characteristics, such as public or private.

With System Security, users identify themselves to the system before performing system functions. Identification is contained in an access code, which can consist of a usercode, password, and charge number combination.

The usercode, password, and charge number access combination is typed in when logging in to the system or entering the BEGINUSER (Beginning Of Job Stream) or USER (Assign Access Code) commands.

After log in, the MCP checks each command entered with the access level associated with that user. If the command is permitted, it is performed, if invalid, an appropriate message is displayed.

See Also:

BEGINUSER (Mark Beginning Of Job Stream); ENDUSER (Mark End Of Job Stream); LO (Log Out); and USER (Assign Access Code) commands.

The *V Series Security Installation and Operations Guide* for information about <usercode>, <password>, and <charge no>.

LN (TRANSFER AND PRINT LOG)

Keyboard Input Message

Access Level: 8

The LN command closes the current system log file, directs logging information into a new log file, and starts the log analysis program. The log analysis program produces a printer backup file containing a report on the log.

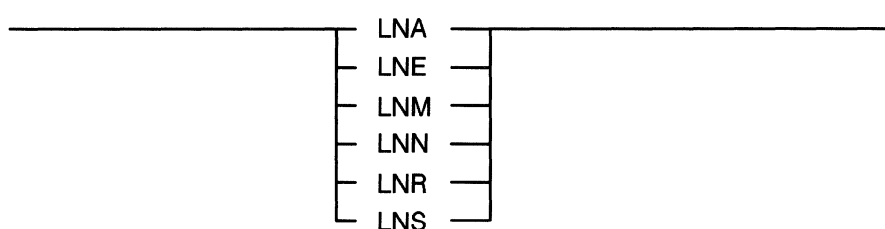


Figure 2-116. LC Command Syntax

■ LNA

transfers and analyzes all log files— Run Log, Maintenance Log, ECM Log, and ODT Log.

■ LNE

transfers and analyzes the ECM Log. The Environmental Control Module Log is created only on V 500 series systems and is stored on the hard disk of the system console.

■ LNM

transfers and analyzes the Maintenance Log.

■ LNN

transfers and analyzes the Network Log.

LNR

transfers and analyzes the Run Log.

LNS

transfers and analyzes the ODT Log.

Use

Log transfer happens automatically when a log file becomes full or at LOGXFERTIME. The LN or TL (Transfer Log) command can be used to close the current log file so you can analyze it or when you want to start a new log file. This depends on the information you are trying to capture.

The LN command transfers the log and starts the log analysis program. LN always starts the log analysis program; TL starts it only if the AUTO option is set in the log's USE record in the system configuration file.

The LN command combines the functions of a TL (Transfer Log) command and an EXECUTE (Execute Library Program) request of the appropriate log analysis program.

Details

The system configuration file can contain a USE record for the RLOG and SLOG (that is, the USE RLOG and USE SLOG records). If a log's USE record is present, the log will be created.

If the AUTO option is set on a log's USE record, the appropriate log analysis and reporting program is started whenever that log is transferred, whether it is transferred automatically when it fills up, at LOGXFERTIME, or manually by the TL or LN command.

If the WRAP option is set on a log's USE record, log files are renamed each time they transfer by incrementing a number in the file name from 0 to 9. For example, the name of the MLOG would change from MLOGs1 to MLOGs9 (where s is the system number). If this option is not set, the file names do not change, and one log file will overwrite the other file at log transfer time. If this option is not used, the files will be renamed MLOGs0 to MLOGs1.

Examples

LNМ

transfers and analyzes the Maintenance log.

LNA

transfers and analyzes all logs.

See Also:

TL (Transfer Log) command.

The USE LOG and LOGXFERTIME records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *V Series System Software Logging Operations Reference Manual* for complete details on system logging.

LO (LOG OUT)

Keyboard Input Message and MCP Control Instruction

Access Level: 0

The LO command logs a peripheral device off the system.

The syntax as shown in Figure 2-117 is the same for either LO command.



— LO —————|

Figure 2-117. LO Command Syntax

See Also:

BEGINUSER (Mark Beginning Of Job Stream); ENDUSER (Mark End Of Job Stream); LI (Log In); and USER (Assign Access Code) commands.

LOAD (Move Files From Library Tape to Disk or Pack)

MCP Control Instruction

Access Level: 5

The LOAD command moves files from library tape to disk or pack and then replaces any existing file with any file on the tape having the same file-identifier.

It is recommended that you use the COPY command instead of the LOAD command. The LOAD command will become unavailable at a future release.

The LOAD command starts a program called LOADMP, if the files are to be moved from disk, or a program called PACKUP, if files are to be moved from pack. LOADMP and PACKUP must be written by the user; Unisys does not supply them. If the appropriate program is not present on disk, the LOAD command is ignored.

See Appendix B for LOAD syntax, if you have user-coded LOADMP and/or PACKUP utilities.

See Also:

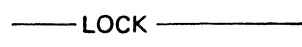
COPY (File Transfer Utility) command. Appendix B.

LOCK (Lock Program)

MCP Control Instruction

Access Level: 1

The LOCK command prevents a job from being discontinued or modified.



— LOCK —|

Figure 2-118. LOCK Command Syntax

Details

The LOCK command can be used with any library maintenance or system command that causes the initiation of a program.

You cannot use the following commands on a locked program:

- DP (Dump And Discontinue)
- DS (Discontinue Program)
- IN (Insert Data Into Program)
- QT (Quit Program Operation)
- RS (Remove Job From Schedule)
- SK (Skip Backup Records)
- SW (Set Programmatic Switches)

To use one of these commands, the program first must be unlocked through the UP (Unlock Program) command.

Examples

EXECUTE PROGA; LOCK
executes and locks a program called PROGA.

PFM LIST FILEA N; CG 27; PR 5; LOCK
schedules the MCP bound intrinsic DMPALL to print the contents of FILEA in a numeric output format; assigns the job a charge number of 27; gives it a memory, I/O and processing, and schedule priority setting of 5; and locks the program.

See Also:

LP (Lock Program) and UP (Unlock Program) commands.

LP (Lock Program)

Keyboard Input Message

Access Level: 1

The LP command locks a specified program, preventing it from being discontinued or modified until it is unlocked by the UP (Unlock Program) command.

— <mix no> LP —|

Figure 2-119. LP Command Syntax

<mix no>

is the mix number of the program you want to lock.

Details

The following commands are invalid for locked programs:

- DS (Discontinue Program)
- DP (Dump and Discontinue)
- IN (Insert Data into Program)
- QT (Quit Program Operation)
- RS (Remove Job From Schedule)
- SK (Skip Backup Records)
- SW (Set Programmatic Switches)

Example

54 LP

locks the job associated with mix number 54.

See Also:

LOCK (Lock Program) and UP (Unlock Program) commands.

MEMDUMP (Abnormal EOJ Memory Dump)

MCP Control Instruction

Access Level: 1

The MEMDUMP command causes program memory to be dumped if the job terminates abnormally. However, the MEMDUMP command is not executed on any jobs terminating because of a memory parity error.

— MEMDUMP —

Figure 2-120. MEMDUMP Command Syntax

Details

If a program experiences a fault or attempts an illegal action upon a file, the MCP will terminate the program. If the program has been executed with the MEMDUMP command or if the TRMD is set, a program fault causes the program to abort with a memory dump, unless termination was due to a memory parity error.

Once the file is dumped, it can be printed and analyzed by using the PM (Print Memory Dump) command.

Example

EXECUTE APROG; MEMDUMP

executes a program called APROG and specifies that program memory will be dumped if the file terminates abnormally.

See also:

DM (Dump And Continue); DP (Dump and Discontinue); and PM (Print Memory Dump) commands.

The USE TRMD record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

MEMORY (Allocate Memory For Program Execution)

MCP Control Instruction

Access Level: 3

When a program is compiled, the amount of memory it needs in order to execute is specified by the programmer or, by default, determined by the compiler. The MEMORY command overrides this requirement.

The MEMORY command is ignored if it specifies less memory than the amount chosen when the program was compiled.

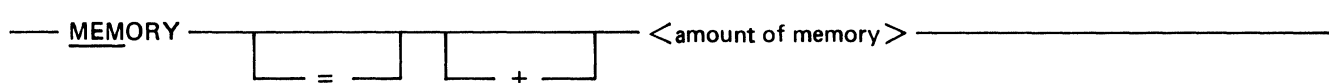


Figure 2-121. MEMORY Command Syntax

MEMORY

MEM

are synonymous and allocate memory for a program's execution.

=

is optional and does not influence the command.

+

adds the specified amount of memory to the default minimum. Multiple MEMORY + <amount of memory> statements are cumulative.

<amount of memory>

indicates the amount of memory required if used without the plus sign. If the <amount of memory> is used with the plus sign, it indicates the amount of memory to add to the default minimum.

If less than four digits are used to indicate the amount of memory, they indicate thousands of digits (KD). That is, 1 equals 1000 digits of memory; 22 equals 22,000 digits of memory; 333 equals 333,000 digits of memory.

If four to six digits are used, they indicate digits of memory. If that figure is not a multiple of 1000, it is changed to the next highest multiple of 1000.

Examples

EXECUTE PAYROL; MEM = 30000

starts a program called PAYROL and specifies that it needs 30000 digits of memory to execute.

PERFORM MTPMTP A 1000 10 B 1000 10; MEM 44

starts the MCP DMPALL utility with 44 thousand digits of memory to perform a media conversion. A file with the file-identifier of A, with an input record length of 1000 bytes, blocking factor of 10, will be copied to a file-identifier of B, a record length of 1000 bytes, and a blocking factor of 10.

COMPILE BTEST PASCAL; MEM + 80

compiles a Pascal program called BTEST, allocating to the Pascal compiler 80,000 digits more than the default minimum amount of memory. (Note that MEM on a COMPILE command does not cause any change in the size of the compiled program.)

MERGE (Transfer Files From Multiple Sources, Resolving Duplicates)

MCP Control Instruction

Access Level: 5

MERGE invokes the SYSTEM/COPY utility to transfer files from any number of sources to up to 10 destinations. Duplicate files are resolved by the MERGE function.

See the COPY (File Transfer Utility) command for details on syntax and how to use this command.

See Also:

COPY (File Transfer Utility) command.

SYSTEM/COPY in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

MOVE (Move Files To New Media)

MCP Control Instruction

Access Level: 5

MOVE invokes the SYSTEM/COPY utility to transfer one or more files from one media to another and remove them from the original media.

See the COPY (File Transfer Utility) command, for details on syntax and how to use this command.

See Also:

COPY (File Transfer Utility) command.

SYSTEM/COPY in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

MR (Remove Duplicate File)

Keyboard Input Message

Access Level: 5

The MR command removes a new file from disk or pack to resolve a DUP LIB situation. The old file will remain intact.

A DUP LIB situation occurs when a program is trying to create a file but a file with that name already exists.

The MR command removes the output file, allowing the "old" file to remain on disk or pack and the program to resume processing.

— <mix no> MR —|

Figure 2-122. MR Command Syntax

<mix no>

is the mix number of the task that is creating the file you want to remove.

Details

The MR command removes the "new" file and leaves the "old" file, while the RM (Remove Duplicate File) command removes the "old" file and leaves the "new" file.

The RMOV option, if set, can resolve DUP LIB conditions on disk or pack by indicating that the old file is to be removed.

Example

The following is a scenario of messages displayed on the ODT in which the MR command is used. (These messages would display on your system only if the OPEN, CLOS, BOJ, and EOJ system options were set.)

ODT Display

Comment

==> EXECUTE JOB1

The operator starts a program called JOB1.

BOJ JOB1=012 012586 9:15

JOB1 starts to execute and is assigned a mix number of 012.

DSK FILE FILEA OPEN OUT JOB1=012

JOB1 opens a file called FILEA. It is a disk file and has been opened for output.

DSK FILE FILEA CLOSED RELEASED
JOB1=12

JOB1 closes the file FILEA.

==> EXECUTE JOB2

The operator starts a program called JOB2.

BOJ JOB2=018 103185 9:20

JOB2 starts to execute and is assigned a mix number of 018.

DSK FILE FILEA OPEN OUT JOB2=018

JOB2 creates a file which happens to be called FILEA and which is a disk file.

**DUP LIB FILEA JOB2=018

JOB2 tries to close the file FILEA. The MCP detects that there is already a file called FILEA. JOB2 waits.

==> SPO 018 MR

The operator enters the MR command. This will remove the version of FILEA that has been created by JOB2. The "old" version of FILEA, created by JOB1, will remain on disk.

EOJ JOB2=018 9:30 IN 1:36 CHG
2:22

JOB2 continues executing until it finishes.

See also:

RM (Remove Duplicate File) and REMOVE (Delete Files) commands.

The USE RMOV record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

MX (DISPLAY JOBS IN MIX)

Keyboard Input Message

Access Level: 0

The MX command displays a list, in priority order, of the jobs currently in the mix. These jobs are active, waiting, or stopped. You can list a subset of the jobs in the mix by using syntax options of the command.

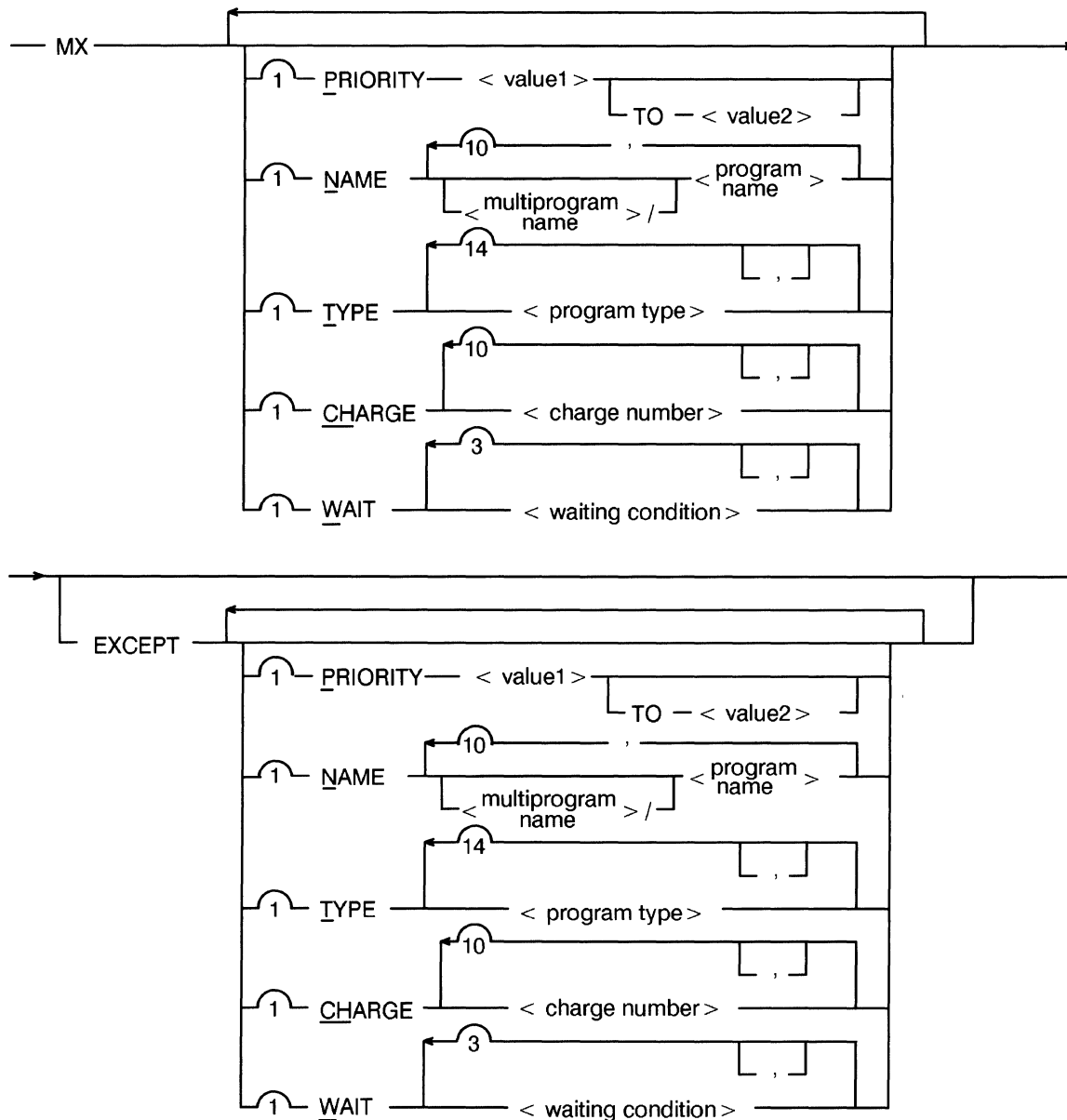


Figure 2-123. MX Command Syntax

■ **PRIORITY** <value1> TO <value2>

Selects jobs based on processor priority. You have the option of selecting a single priority or a range of priorities. <value1> and the optional <value2> must be integers between 1 to 9. <value1> must be less than or equal to <value2>. The abbreviation P is accepted for PRIORITY.

■ **NAME** <multiprogram name> / <program name>

Selects jobs based on the program name and (optionally) on the multiprogram name. You can enter a maximum of 10 name selections, separated by commas. The abbreviation N is accepted for NAME.

You can also select jobs based on part of a name by masking the name using the equal sign as a masking character. Use one equal sign for each unknown character position. For example,

MX NAME DMP=== EXCEPT NAME DMPOUT

displays all programs in the mix beginning with the three characters DMP, but does not display the program DMPOUT.

■ **TYPE** <program type>

Selects jobs based on the type of program. The abbreviation T is accepted for TYPE. You can enter a maximum of 14 type selections. The available program types are:

EXECUTE	Program is executing
COMPILER	Program generates object code
DMPALL	Program is DMPALL intrinsic
LOADMP	Program is LOADMP intrinsic
PACKUP	Program is PACKUP intrinsic
DSKOUT	Program is DSKOUT intrinsic or diskpack SQUASH intrinsic
MCS	Program is Message Control System
TIMESHARE	Program is executed as time sharing task
SHAREHNDL	Program is a time sharing handler program (for example, CANDE)
HANDLERS	Program is any type of handler (BNA, WFL, time sharing)
COMPCHAR	Program that generates object code, executed as a timesharing task
DMSPROG	Program is a DMSII control program (for example, DBP)
WFLHNDL	Program is a Work Flow Language (WFL) handler
BNAHNDL	Program is a BNA network architecture handler
COPY	Program is SYSTEM/COPY intrinsic
SORT	Program is SORT: or SORT. intrinsic
UTILITY	Program is any type of intrinsic (DMPALL, LOADMP, PACKUP, DSKOUT, SYSTEM/COPY)

CHARGE <charge number>

Selects jobs based on the charge number the job is executed under. You can enter as many as 10 charge number selections and 10 charge number exceptions. The abbreviation CH is accepted for CHARGE.

You can also select jobs based on part of a charge number by masking the charge number using the equal sign as a masking character. Use one equal sign for each unknown character position. For example,

MX CH 66==== EXCEPT CH 660001, 669999

displays jobs executed under a charge number beginning with the characters 66, but does not display jobs executed under charge numbers 660001 or 669999.

WAIT <waiting condition>

Selects jobs that are waiting for a specific condition to be satisfied (for example, waiting hardware, waiting memory and so forth). You can enter as many as 3 waiting conditions and 3 waiting condition exceptions. The abbreviation W is accepted for WAIT.

The available types of waiting conditions are:

<u>SPACE</u>	Waiting for memory, disk space or diskpack space
<u>DEVICE</u>	Waiting for hardware device to become available
<u>OPERATOR</u>	Waiting for operator to perform an action
<u>EVENT</u>	Waiting for a software event to be signalled (for example, STOQUE, CRCR and so forth)

EXCEPT

All text following the keyword EXCEPT is treated as exceptions to the selections entered previously. All of the exceptions must be grouped together after the keyword EXCEPT; you cannot enter a selection and an exception followed by another selection.

See Also:

AJ (Display Active Jobs); WJ (Display Waiting Jobs); and WS (Display Schedule) commands.

NET (NETWORK CONTROL COMMAND)

Keyboard Input Message

Access Level: 8

The NET command displays the status of a BNA network, or changes the mode of the local system to or from the Network mode.

To Display Network Status:

NET

Figure 2-124. NET Command Syntax (Display Status)

The system response to this command is shown in Figure 2-125.

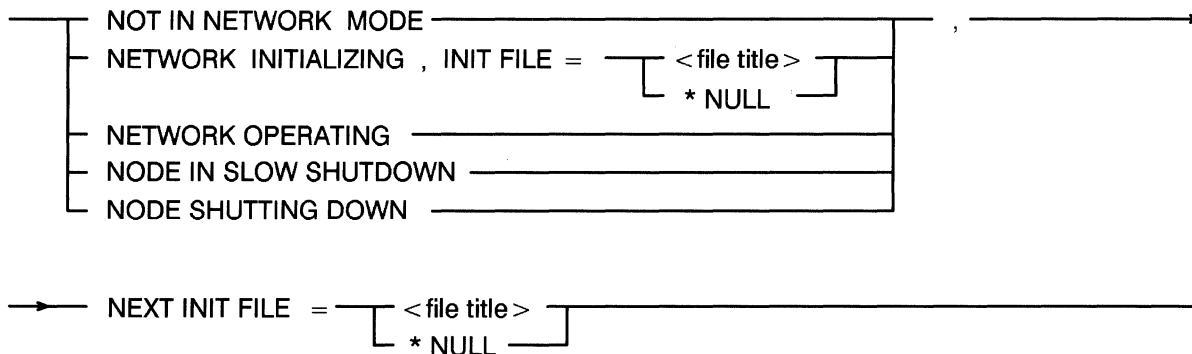


Figure 2-125. Response to NET Command

To Change Network Mode

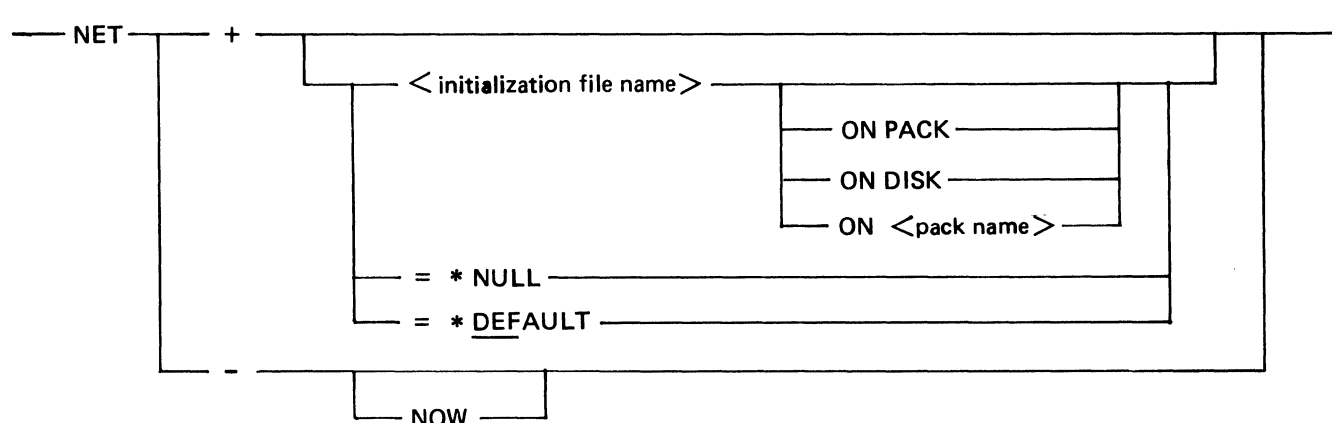


Figure 2-126. NET Command Syntax (Change Network Mode)

Next Initialization File

An initialization file, or "init" file, contains a series of BNA commands that bring the local system into the network.

At any one time, one init file is designated as the "next init file." When a NET + command is done, the BNA commands for bringing up the network are taken from the current "next init file."

- + puts the local system into Network mode, using the BNA commands in the next initialization file.
- + <initialization file name>
puts the local system into Network mode, using the BNA commands in the named initialization file.
- puts the local system in Isolated mode, taking it out of the network. This takes effect after all subports that were in communication with remote ports finish their dialogs and are deactivated.

ON PACK

ON DISK

ON <pack name>

gives the location of the specified initialization file.

= *DEFAULT

makes the file BNAINI on pack the next initialization file.

= *NULL

instructs the system to have no next initialization file. When the network is brought up, the operator will have to enter BNA commands from the ODT.

-NOW

puts the local system in Isolated mode immediately, deactivating all subports that were in communication with remote subports without waiting for dialogs to finish.

V Series MCP/VS System Software Operations Guide, Volume 2
The System Commands

Examples

NET +
puts the local system in Network mode, using the next initialization file.

NET + INITFI ON DISK
puts the local system in Network mode, using the initialization file INITFI on disk.

See also:

The V Series BNA Installation And Operations Guide.

NL (Resume At New Location)

Keyboard Input Message

Access Level: 3

The NL command resumes the processing of a program that has been suspended (awaiting DS or DP).

— <mix no> NL <program address> —

Figure 2-127. NL Command Syntax.

<mix no>

designates the program that is to be resumed. You cannot use a mix number of 0 or 1.

<program address>

is the base-relative address at which point the processing of the program is to resume. It must be an even value within bound of the suspended program.

CAUTION

This command should be used only when the reason for suspension of the program has been thoroughly investigated. The MCP has no means to determine whether the new location is acceptable or other possible conflicts. The MCP can determine only that the address is within the program's base and limit.

Example

10 NL 15532

resumes processing of the program associated with mix number 10 at program address 15532.

See also:

ST (Suspend Program Processing) and STOP (Suspend Program Processing) commands.

The USE TERM and USE TRMD records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

NT (Terminate Trace)

Keyboard Input Message

Access Level: 3

The NT command manually stops the instruction trace of a program.

——<mix no> NT——|

Figure 2-128. NT Command Syntax

<mix no>

is the mix number of the program being traced.

Details

Instruction traces can be started with a GT (Initiate Trace) command if the program is executing, with a TRACE (Trace A Program) command prior to a program's execution, or programmatically through a TRACE branch communicate (BCT). The TRACE BCT can also be used to terminate a program trace.

Example

33 NT

stops the tracing of the program with mix number 033.

See Also:

GT (Initiate Trace) and TRACE (Trace A Program) commands.

The *V Series Program Interfaces Programming Reference Manual* and the various program language manuals for information about TRACE.

NW (Networking Command Prefix)

Keyboard Input Message

Access Level: 8

Most BNA network architecture commands must be prefaced by NW.

— NW <networking command> —|

Figure 2-129. NW Command Syntax

<networking command>

See the *V Series BNA Installation and Operations Guide* for full details on BNA commands.

Exceptions

The BNA commands that must not be prefaced by NW are:

- AT
- HN
- NET

An ODT cannot be dedicated to BNA by the DEDICATED command. The NW prefix is not used from an ODT that is dedicated to BNA.

Example

NW AWAITCALL BY JMSTATN

places the specified station called JMSTATN into a state to receive a call.

See also:

The *V Series BNA Installation and Operations Guide*.

OF (Optional File Indication)

Keyboard Input Message

Access Level: 1

The OF command is used to respond to a NO FILE, FILE NOT ON DSK, or FILE NOT ON DPK message. It causes the waiting program to continue without the file. The requested file must have been declared by the programmer as being an optional file.

— <mix no> OF —

Figure 2-130. OF Command Syntax

<mix no>

is the mix number of the program you wish to continue executing.

Use

This command is used in the following situation:

1. A job suspends its execution and waits because a file it needs is missing. This condition produces a NO FILE, FILE NOT ON DSK, or FILE NOT ON DPK message on the ODT.
2. The missing file is one that the programmer has declared OPTIONAL, meaning that it is not needed every time the program runs.
3. The OF command causes the program to continue executing, ignoring the missing file.

Example

12 OF

informs the MCP that the optional file being sought can be ignored and the program with mix number 012 can continue to execute.

See Also:

DP (Dump And Discontinue); DS (Discontinue Program); IL (Assign Labeled File); OK (Continue Processing Suspended Program); and UL (Assign Unlabeled File) commands.

The various programming language reference manuals for information about declaring optional files.

OK (Continue Processing Suspended Program)

Keyboard Input Message

Access Level: 1

The OK command directs the MCP to continue processing a suspended program. The program might be suspended because of a lack of resources or a resource conflict.

The OK command can be used after the operator has resolved a no-user-disk condition. In most cases, the MCP automatically recognizes that the program need no longer wait and resumes processing without operator intervention.

Do not initiate the OK command until the problem has been resolved.

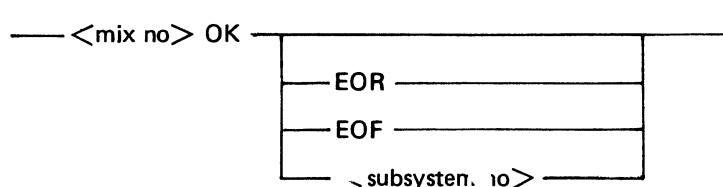


Figure 2-131. OK Command Syntax

<mix no>
is the mix number of the suspended program.

EOR

EOF

can be used when an input file on magnetic tape has no ending label or has an irrecoverable parity error on the ending label.

The EOR option sends the job an End-Of-Reel message and causes the job to continue.

The EOF option sends the job an End-Of File message and causes the job to continue.

<subsystem no>

can be used only if the specified program is in a NO USER DSK condition because insufficient space is available on the disk subsystem indicated in the ODT message. The operator can cause the remainder of the file to be allocated on a different subsystem by including the <subsystem no>, which can have a value from 0 to 8. A value of 0 indicates the default subsystem.

Examples

3 OK

directs the MCP to continue processing the program with mix number 003.

35 OK 2

directs the MCP to continue processing the program with mix number 35, which lacks sufficient space on its disk subsystem. The MCP will try to allocate the needed file space on subsystem 2.

18 OK EOR

directs the MCP to continue processing the program with mix number 018 by sending the program an End-Of-Reel message.

See also:

FR (Designate Final Reel); IL (Assign Labeled File); OU (Direct File To Device); RM (Remove Duplicate File); and UL (Assign Unlabeled File) commands.

OL (Display Peripheral Status)

Keyboard Input Message

Access Level: 0

The OL command displays the status of a channel, peripheral unit or group of peripheral units, or pseudo card readers.

Channels

You can display the status of a channel by using the syntax shown in Figure 2-132.

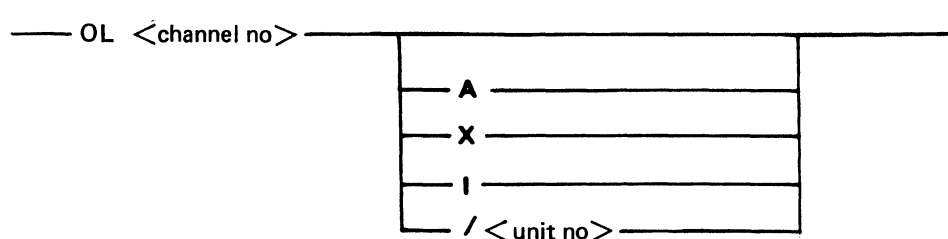


Figure 2-132. OL Command Syntax

<channel no>

is the channel whose status will be displayed. If only the <channel no> is supplied, the MCP displays the status of that channel.

A

displays the status of all units on the specified channel.

X

displays the status of all unassigned units on the specified channel.

I

displays the status of all assigned units in use on the specified channel.

/ <unit no>

obtains the status of a specific unit.

Examples

OL 2

displays the status of channel 2.

OL 3/2

displays the status of channel 3, unit number 2.

OL 6A

displays the status of all units declared for channel 6.

Peripheral Units

You can display information about a group of hardware devices by using the syntax shown in Figure 2-133.

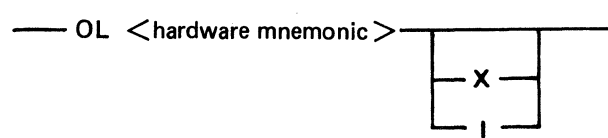


Figure 2-133. OL Command Syntax (Hardware Type)

<hardware mnemonic>

displays the status of all devices of the specified <hardware mnemonic> if the optional elements are not used.

The <hardware mnemonic>s permitted are listed in Table 2-11. These mnemonics resemble those specified in the UNIT record of the system configuration file. In addition, the identifier MTP causes the MCP to list the status of all magnetic tape units.

Table 2-11. Hardware Mnemonics

CRD	80-column card reader
DCP	Data communications Processor
GCR	GCR magnetic tape drive
IPP	Intelligent Laser Printing System
ISC	Inter-System Connect
MPE	Phase-Encoded (PE) magnetic tape drive
MT9	NonReturn Zero magnetic tape drive
MTC	Magnetic cartridge tape drive
NCP	Network Communications Port
NST	Non-Status device
OCS	Operator Control Station
ODT	Operator Display Terminal
PCH	80-column card punch
PRN	buffered printer
RJE	Remote Job Entry device
S4A	9137 type reader sorter
S4B	9138 type reader sorter
TC5	TC 500 device
TWX	teletypewriter through a UNILINE DLP
TPR	train printer
VDD	B 20 or ET series workstation or a B 9352 or TD 800 series terminal

X

displays the status of all unassigned units of the designated <hardware mnemonic>.

I

displays the status of the units of the designated <hardware mnemonic> that are being used.

Sample Responses

If you try to display the status of a magnetic tape unit that is in use, you will receive one of the following messages:

Message	Meaning
IN USE (I)	the tape unit is opened input
IN USE (O)	the tape unit is opened output
IN USE	the tape unit is closed and still attached to the job

The OL command for disk and pack can be OL DPK, OL DSK, or OL <cc/u>. The first two commands show information for all disks or packs; the latter command displays information for a particular device. The response to an OL command for disk or pack is illustrated in Figure 2-134.

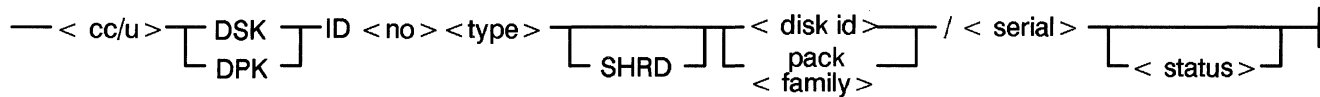


Figure 2-134. OL Command Response (Disk/Diskpack)

<cc/u>

indicates the channel and unit number of the device.

DSK
DPK

indicates disk or pack, respectively.

ID <no>

is the ID number of the device.

<type>

indicates the type of device, such as 5N, 235, 206I, 207I. A response of “UNKN” (unknown) indicates that the device has not been successfully classified.

SHRD

indicates a shared device.

<disk id>

is the disk subsystem identifier, which appears in the form of ID<disk id> <subsystem no>. For example, ID0011.

<pack family name>

is the family name for a 180-byte pack.

/<serial>

is the serial number of the device.

<status>

The system will respond with the following message:

<nnnnn> SEGS AVAIL, <utilization> %I/O IN USE BY <nn> FILES

SEGS AVAIL

refers to the total number of available sectors on the device.

NOTE

This is only the total space available on the drive and does not take into account any checkerboarding. Therefore, although it appears from this value that a file should fit on the device, there may actually be insufficient contiguous space for the various areas.

<utilization>

is an estimate of the percentage of time that the device is in use. It is recomputed every 10 seconds in the following manner: the elapsed times between the execution of each Initiate I/O (IIO) instruction and the completion of that I/O are totalled, giving the approximate total time that the unit is in use during the 10-second period; the total time is then expressed as a percentage of a 10-second interval.

IN USE BY

indicates the number of file areas open on the device, not necessarily the number of files.

If a tape unit is GCR/PE capable, the response to an OL MTP command also displays the density of the tape. For example:

06/1 GCR (1600) Y85W4/FILE 8 RL# 001 (I) SYSTEM/COPY=41

The density of the tape is 1600.

Examples

OL MTP

displays the status of all magnetic tape units.

OL SOR I

displays the status of the in-use or assigned sorter units.

OL CRD X

displays the status of all unassigned card readers.

Pseudo Card Readers

You can display the active pseudo card readers on the system by using the syntax shown in Figure 2-135. for active pseudo card readers.

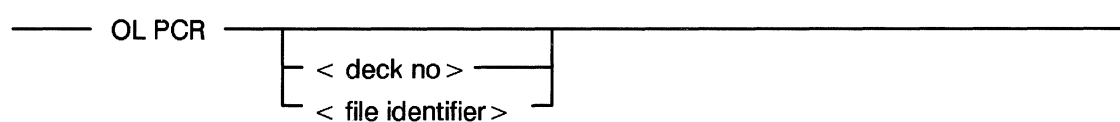


Figure 2-135. OL PCR Command Syntax

PCR

displays the status of all pseudo card readers if the optional elements are not specified.

<deck no>

displays the status of only the indicated pseudo card deck.

<file identifier>

displays the status of all pseudo card readers that contain a deck of the specified name.

Details

Use the CD, CDP (List Inactive Pseudo Decks on Disk or Pack) command to display inactive pseudo decks.

If you want to deactivate a pseudo card deck, use the DA, DAP (Deactivate Pseudo Reader) command.

Examples

OL PCR CARDFL

displays the status of all pseudo card readers that contain a deck with the name CARDFL.

OL PCR 3

displays the status of pseudo card readers with an active deck number 3.

See Also:

CD, CDP (List Pseudo Packs); DA, DAP (Deactivate Pseudo Reader); SHOW (Display System Information) commands.

Network Communication Ports

You can display status information on any Network Communication Ports (NCPs) on the system. NCPs are used as part of the V Series Communication System (VCS). There is no special syntax for the OL NCP command. The response to the OL NCP command is shown in figure 2-135a.

NCP Entry

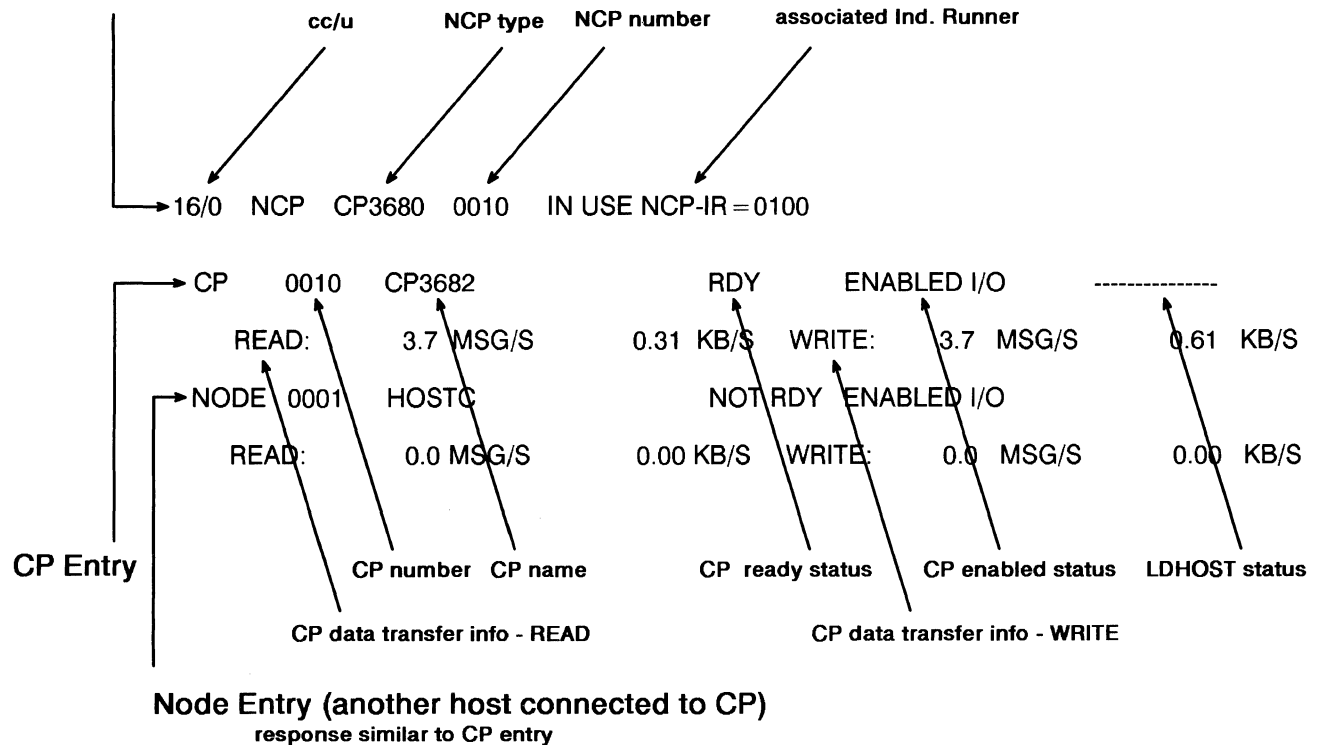


Figure 2-135a. OL NCP Command Response

Entries in the OL NCP response are displayed in the order that items are declared in the VCS initialization file.

There are three types of entries in the response: NCP entry, CP entry and node entry. NCP entries represent the NCPs connected to the system. Communication Processor (CP) entries represent hardware devices with physical connections to individual stations. Currently, there is one CP entry for each NCP entry. Node entries represent other host systems that may be connected to the NCP. There can be several node entries for each NCP entry.

cc/u

is the channel number and unit number declared for the NCP.

NCP type

a mnemonic representing the type of NCP connected to the channel. Current NCP types include CP3680 and TELCOM.

■ NCP number

a 4-digit number assigned to the NCP in the VCS database. This number is declared on the UNIT NCP record in the system configuration file.

■ associated Ind. Runner

status, name and task number of the Independent Runner (IR) responsible for serving the NCP. This item is not displayed if VCS is not currently running on the system.

■ CP number

a 4-digit number assigned to the CP in the VCS database.

■ CP name

a 6-character name assigned to the CP in the VCS database.

■ CP ready status

indicates whether the CP is ready.

■ CP enabled status

indicates what input/output mode the CP is currently enabled for. Possible values include ENABLED I/O, ENABLED IN, and ENABLED OUT.

■ LDHOST status

indicates if the CP needs to be loaded with firmware. If the CP is currently loaded with the correct firmware, this field is blank. If the words "LDHOST REQD" appear, the LH command (LH NCP option) must be used to load the appropriate firmware to the CP.

■ CP data transfer info— READ

■ CP data transfer info— WRITE

CP status information showing the messages per second and the kilobytes per second transmitted through the CP for both read and write operations. The averages are calculated when the OL NCP command is entered, based on the period of time since the CP status (status interval 40 seconds).

■ Details

To calculate the total bandpass of an NCP, add the read and write transfer information for all CPs and nodes connected to the NCP. The result can be compared to the maximum bandpass of the NCP for performance capacity planning.

■ See Also:

■ UNIT (Add Peripheral Devices) command

■ NCP initialization information in the *V Series Communication System Initialization Reference Manual*.

O

The OT command displays on the ODT up to 30 characters of a job's memory.



specifies the program from which the data is displayed. You cannot have a mix number of 0 or 1, which are reserved for the use of the operating system.

designates the program base-relative location at which the data begins.

denotes the number of bytes (for UA data) or digits (for UN data) to be displayed and must be 30 or less.

indicate data class.

UA denotes an unsigned alphanumeric. This causes data to be displayed as characters with the <length> in bytes. (One byte is 8 bits.)

UN is unsigned numeric. This causes data to be displayed as a string of digits with the <length> in digits. (One digit is 4 bits.)

11 OT 38 2 UN

will display a field two unsigned numeric digits long from the program with mix number 011 at base-relative address 38.

OU (Direct File To Device)

Keyboard Input Message

Access Level: 5

The OU command routes an output file to a designated device. Within limits, this command can be used to change the hardware medium for a file.

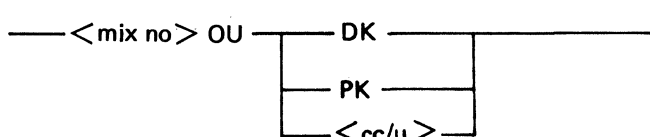


Figure 2-137. OU Command Syntax

<mix no>

must specify a program that requires an output device. The <mix no> cannot be 0 or 1, which indicates the MCP.

DK

routes the file to disk.

PK

routes the file to pack.

<cc/u>

routes the file to the specified channel number and unit number of a magnetic tape drive. The device specified by <cc/u> cannot be in a saved status.

Use

The OU command is normally used in response to a "<hardware mnemonic> REQD <file name> <job name> = <mix no>" message that has been generated because of the following:

- a lack of available resources
- system option settings that do not permit automatic file assignment

The system options that affect file assignment, such as designating the medium where printer backup files will be directed, include APBD, APCR, PBD, PBP, PBPS, PBT, PCD, PCP, and PRN. These options can be set through the USE records of the system configuration file or with the SO (Set System Options) command. These options can be reset with the RO (Reset System Options) command.

Details

If the PBT option is set, a file directed to a GCR (Group Coded Recording) tape drive will be written in GCR mode if a scratch tape is mounted. If the tape mounted is not a scratch tape, the already established mode and density are used.

Examples

15 OU 3/2

sends an output file with mix number 015 to the tape drive with channel number 3, unit number 2.

20 OU DK

sends an output file with mix number 020 to disk.

See also:

FM (Special Forms Response); RO (Reset System Options); and SO (Set System Options) commands.

The USE records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

PA (ANALYZE PACK DIRECTORY)

Keyboard Input Message

Access Level: 5

The PA command invokes the intrinsic program DSKOUT to generate an analysis of all files found in the pack directory. An analysis of the following is given:

- area addresses
- file identifier
- total number of segments used
- logical record size
- records per block
- area size
- area numbers, total areas used, and maximum areas declared
- end of file pointer
- file attributes

You can specify a single file id or group file id, and the system analyzes only the relevant file(s).

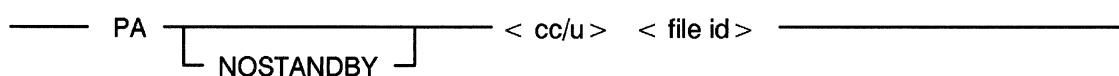


Figure 2-138. PA Command Syntax

■ NOSTANDBY

directs the operating system to capture information for DSKOUT even when the operating system *cannot* perform its normal standby function. The standby function is still attempted, but information is captured regardless of whether the function is successful. For more information about the standby function, see Details on the following pages.

<cc/u>

is the channel number/unit number of the pack to be analyzed.

<file id>

is the file identifier or masked file identifier of files on which an analysis will be performed.

Output Options

Output options that provide alternate sorting and analysis are available. See the KA (Analyze Disk Directory) command for a full description of these options and their applications.

Details

When you enter a PA command, the operating system can respond with the message:

KBD IGNORED: DISK OR PACK MAINT IS IN PROGRESS

This means the operating system cannot perform a standby function as part of the PA command. A standby function suspends updates to the pack directory and other data structures long enough to obtain an accurate "snapshot" of the structures. This "snapshot" is passed to DSKOUT for analysis. The operating system needs exclusive access to the pack directory for an accurate DSKOUT analysis.

To respond to this situation, you can:

- Wait until a later time when pack activity is less
- Use the NOSTANDBY option of the PA command

NOSTANDBY captures information for DSKOUT even when the operating system *cannot* perform the standby function. The standby function is attempted, but information is captured regardless of whether it is successful.

Using NOSTANDBY can create missing or duplicate pack messages that are not correct. This happens if an update occurs as the "snapshot" is taken. If NOSTANDBY is used and missing/duplicate pack conditions are encountered, DSKOUT flags the resulting messages with NOSTANDBY warnings.

These warnings mean the condition may be caused by non-exclusive access rather than actual missing or duplicate pack space. Before you act on any missing/duplicate pack messages with a NOSTANDBY warning, enter another PA command without NOSTANDBY. Enter the command when less pack activity is taking place.

Examples

PA 10/12 ABC=D=

produces an analysis of all files with the masked file id ABC=D= on the pack located on channel number 10, unit 12.

PA 3/1 AFILE

produces an analysis of AFILE on channel number 3, unit 1.

PA NOSTANDBY 14/1 =

produces an analysis of all files on the pack located on channel number 14, unit 1. The analysis is generated even if the normal standby function of the PA command is unsuccessful.

See Also:

KA (Analyze Disk Directory); KP (Print Disk Segments); KS (Analyze Disk Space); PP (Print Pack Sectors); and PS (Analyze Pack Space) commands.



PACK (Add A Pack)

MCP Control Instruction

Access Level: 8

The PACK command adds a pack to the system environment.

The syntax for the PACK command is identical to that for the PACK record of the system configuration file.

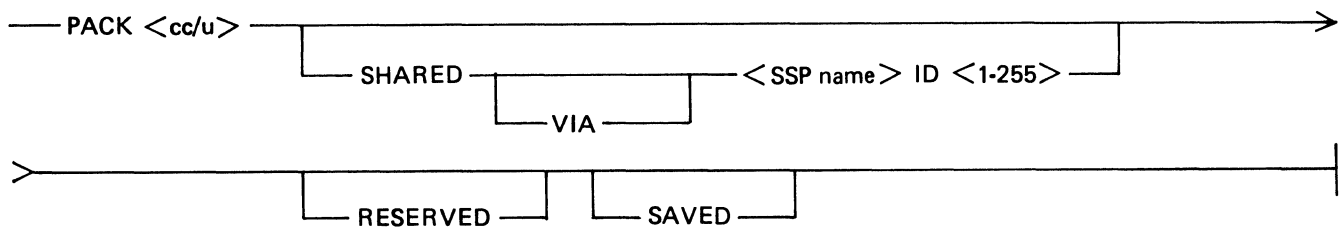


Figure 2-139. PACK Command Syntax

<cc/u>

is the channel number/unit number of the pack to be added.

ID <1 - 255>

assigns the pack an ID number. Do not use the same ID number twice on a given system. Be sure that each system in a multi-system shared configuration uses the same ID number to reference the drive.

SHARED <SSP name>

SHARED VIA <SSP name>

indicates the name of the SSP that connects the system to other systems in a multi-system shared configuration.

RESERVED

causes the device to be initially RESERVED when the system is cold started or at subsequent halt/loads. RESERVED status can be altered using the UR (Inhibit or Uninhibit Unit) command.

SAVED

causes the device to be initially SAVED when the system is cold started or at subsequent halt/loads. The device must be readied with the RY (Ready Peripheral Device) command in order to access it. Note that the RY command does NOT alter this initial SAVED status.

V Series MCP/VS System Software Operations Guide, Volume 2
The System Commands

Example

PACK 05/2 ID 3 SAVED

adds a pack to the system environment on channel 5, unit number 2. The pack has an ID number of 3, and after a cold start will have a saved status until readied with an RY command.

See Also:

DISK (Add A Disk) and UNIT (Add Peripheral Devices) commands.

The PACK record in *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

PATCH (Patch File)

MCP Control Instruction

Access Level: 5

The PATCH command overwrites a specified section of data with new data. The area can be in an MCP code file, a program code file, in a disk or pack file, or at a known sector on a disk.

Note that MCP code files can reside only on disk. Any attempt to patch such a file on pack will result in a syntax error.

Patches to files on pack will be allowed only if the relevant area resides on the base pack. Continuation pack areas cannot be patched or displayed.

Patching A Running MCP Code File

Option 1 patches the running MCP code file.

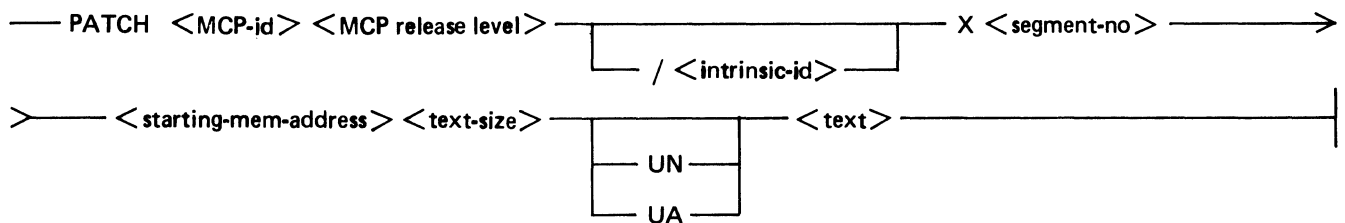


Figure 2-140. PATCH Command Syntax (Running MCP Code File)

Patching Program or Non-Running Code Files

Option 2 patches a program or a non-running code file.

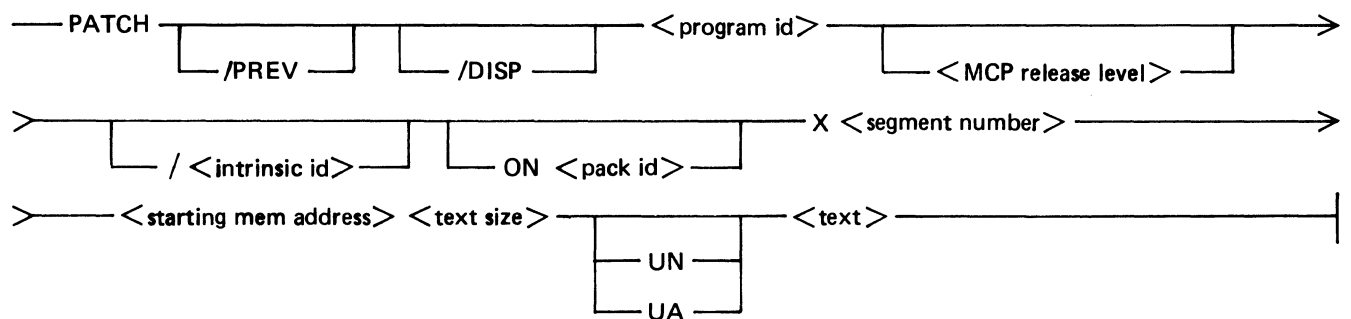


Figure 2-141. PATCH Command Syntax (Program or Non-Running Code File)

Patching Disk or Pack Files

Option 3 patches a file on disk or pack.

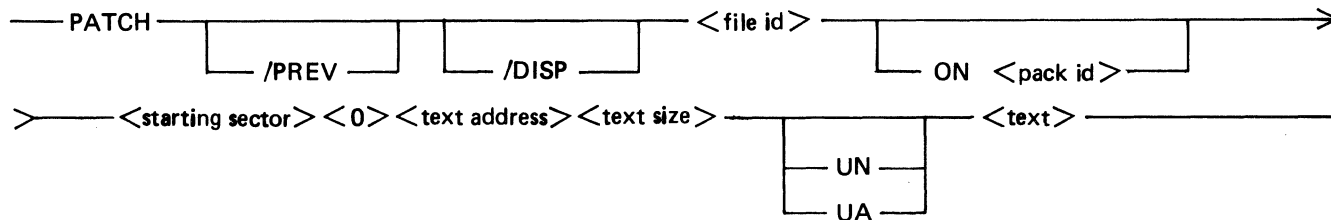


Figure 2-142. PATCH Command Syntax (File)

Patching Disk Sectors

Option 4 patches a disk sector.

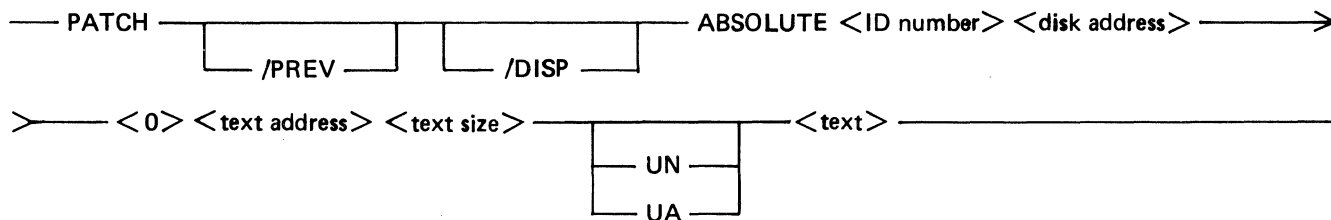


Figure 2-143. PATCH Command Syntax (Disk Sector)

NOTE

The text given must be the specified length or a syntax error will result. If UN <text> is specified, then only numbers and the hexadecimal characters 0-F are allowed. If UA <text> is specified, and any special characters are present, the entire string must be enclosed in quotes. All PATCHes are logged in the RLOG.

<MCP-id>

specifies the id of the running MCP code file.

/PREV

displays the data present at the location specified before the patch was applied.

/DISP

displays the data after the patch is applied to ensure correctness.

<program id>

is the name of the program to be patched.

<MCP release level>

is the MCP/VS release number.

/<intrinsic id>

is valid only if <program id> specifies MCP and is the name of the MCP intrinsic to be patched.

ON <pack id>

is the name of the pack where the program is located. A pack id of DISK is the same as omitting the ON <pack id> clause.

X <segment number>

is the actual segment where the data is located. Valid numbers are 0-999.

<starting mem address>

is the actual memory address within the segment. This term must be decimal digits.

<text size>

is the amount of data to be patched. The maximum size is 30 digits or bytes.

UN

UA

indicates the format of the data to be patched. UN denotes numeric, UA denotes alphanumeric. If the text is declared as UN, the record can contain only 0-9 and A-F.

The default is UN.

<file id>

is the name of the file to be patched.

<starting sector>

is the sector address of data location. The value is relative to the file being patched and to zero.

0

separates the starting sector from the text address.

<text address>

must be less than 200 for disk or 360 for pack. This address is always expressed in digits, relative to zero.

For Option 3, this causes zero-relative digit displacement from the beginning of the sector.

For Option 4:, this causes displacement from the beginning of the disk sector being patched.

<id number>

is the logical id number of the device and can be any number from 1-255.

<disk address>

is a 1- through 12-digit address representing the physical disk address (in decimal).

Example

PATCH /DISP ABSOLUTE 4 367299 0 100 5 UN 33251

patches a disk sector on the disk with ID number 4 at disk address 367299 with UN data 33251 and displays the data after the patch is applied.

Verifying Patch Records

You can verify patch records by using the PATCHK program. This applies to patching running MCP code files (Option 1), program and non-running MCP code files (Option 2), and patching disk or pack files (Option 3).

To utilize PATCHK, you must create patch records on cards or on card images through an on-line editor such as CANDE or DMPALL/CTLDCK.

Records consist of 80 bytes; the blocking factor can be any value. Only the first 72 bytes of text will be read. Do not enter an invalid punch in column one, or a question mark in column one of a CANDE file that is to become a pseudo deck, until after PATCHK has verified the records.

Copy the check sum from the printed flash supplied by Product Assurance and Support onto the card. PATCHK reads the patch records and calculates a check sum. Errors are detected by comparing the two check sums. PATCHK prints a report listing the errors.

PATCHK uses only the following terms to calculate the check sum:

<program id>
<MCP release level>
<intrinsic id>

If this term is enclosed in parentheses, the parentheses are not included in the check sum.

ON <pack id>
This entire term is included in the check sum.

X <segment no>
PATCHK includes the X in the check sum. Leading zeros are ignored; for example, X05 is equal to X5. However, X00 is equal to X0.

<starting mem address>
Leading zeros are ignored; however, if this field is zero, zero will be included in the check sum.

<starting sector>
Leading zeros are ignored; however, if this field is zero, zero will be included in the check sum.

0
This literal is included in the check sum as a single digit, regardless of additional leading zeros.

<text address>
Leading zeros are ignored; however, if this field is zero, zero will be included in the check sum.

UN
UA

The term UA is accumulated in the check sum; the term UN is not.

All characters entered are included in the check sum, regardless of field type.

If the length of the <text> is not equal to <text size>, the record is flagged to be in error.

If the PATCHK records are in a CANDE file, you can execute PATCHK as follows:



Figure 2-144. Executing PATCHK

is the 4-character id of the CANDE file.

is the first four digits of the CANDE usercode.

A card deck containing invalid punches in column one can be submitted to PATCHK as follows:

C

■ PATCHK Program Error Messages

■ These paragraphs list the PATCHK error messages and provide a sample patch record showing each error.

■ UNKNOWN DATA AFTER "/"

The PATCH option entered after the slash is not a valid PATCH option.

PATCH/PREB MCP 2.02 X23 24610 8 32F00000.9F

■ FILE NAME > 6 CHARS

The name of the program or operating system code file is longer than the maximum 6 characters.

PATCH MCPMCPMCP 2.02 X23 24610 8 32F00000.9F

■ RELEASE LEVEL > 4 CHARS

The release level of the operating system code file is longer than the maximum 4 characters.

PATCH MCP 2.021 X23 24610 8 32F00000.9F

■ SEGMENT NO. LENGTH > 3 OR < 1

The segment number of a program or operating system code file.

PATCH MCP 2.02 X2345 24610 8 32F00000.9F

■ MEMORY AREA NOT PRESENT

The memory area subscript of the segment number is missing.

PATCH MCP 2.02 X23. 24610 8 32F00000.9F

■ MEMORY AREA NOT NUMERIC

The memory area subscript of the segment number contains a non-numeric character.

PATCH MCP 2.02 X23.F 24610 8 32F000000.9F

■ MEMORY AREA > 99

The memory area subscript of the segment number is greater than 99.

PATCH MCP 2.02 X23.120 24610 8 32F000000.9F

■ DISK ADDRESS/SECTOR # NOT NUMERIC

The disk address to be patched contains a non-numeric character.

PATCH ABSOLUTE 1 120F0 0 400000 6 UA HAHAA

■ DATA AFTER "ABSOLUTE" NOT NUMERIC

The ID number, disk address, text address or text size contains a non-numeric character.

PATCH ABSOLUTE F 12000 0 400000 6 UA HAHAA

■ ZERO LITERAL NOT PRESENT

The character zero (0) is missing from a PATCH command that patches a file or a disk sector.

PATCH GUESS 12000 400000 6 UA HAHAA

■ MEMORY ADDRESS NOT NUMERIC

The starting memory address contains a non-numeric character.

PATCH MCP 2.02 X23 2461D 8 32F00000.9F

TEXT LENGTH NOT NUMERIC

The text size contains a non-numeric character.

PATCH MCP 2.02 X23 24610 A 32F00000.9F

TEXT LENGTH <1 OR > 30

The text size entered is not within the limits allowed by the PATCH command. Usually the text size is too large.

PATCH MCP 2.02 X23 24610 124 32F00000.9F

"UA" NOT SPECIFIED

The text to be patched contains a non-numeric character, but the PATCH command does not specify UA data.

PATCH MCP 2.02 X23 24610 8 _ 32X00000.9F

DATA NOT LENGTH SPECIFIED

The text entered in the patch record does not match the text length in the PATCH command.

PATCH MCP 2.02 X23 24610 6 32F00000.9F

INVALID CHECKSUM CHK = 77

An incorrect character has been entered in the patch record.

PATCH MCP 2.02 X23 24610 8 32F00000.CE

See also:

The PATCH record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

PB (PRINT BACKUP FILE)

Keyboard Input Message

Access Level: 1

The PB command starts a user-provided program to print backup files. However, it is recommended that you use the PRINT (Print Files) command instead of the PB command.

If the printer backup file resides on tape, the program PBTOUT is scheduled. If the file resides on disk or pack, PBDOUT is scheduled. PBTOUT and PBDOUT must be written by the user; Unisys does not supply them. If the appropriate program is not present on disk, the PB command is ignored.

See Appendix B for PB syntax.

See also:

Print (Print Files) command. Appendix B.

PC, PCP (Create Punch Backup File)

Keyboard Input Message

Access Level: 1

The PC command schedules the execution of the PCHOUT utility, which creates punched output of disk or pack punch backup files.

You can specify either an MCP or user-coded version of PCHOUT. Figure 2-145 shows the syntax for an MCP version; Figure 2-146 for a user-coded version.

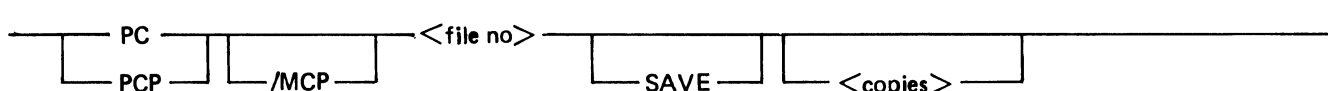


Figure 2-145. PC, PCP Command Syntax (MCP Version)

PC
PCP

PC punches backup files from disk, PCP from pack.

/MCP

specifies the MCP intrinsic PCHOUT.

If you specify /MCP, the bound MCP intrinsic is scheduled regardless of the presence of a user-provided PCHOUT on disk.

If you omit this option and there is a user-provided PCHOUT on disk, the user-provided PCHOUT will be executed.

<file no>

is the number of the punch backup file on disk or pack to be made into punched card output.

SAVE

indicates that the punch backup file is not to be removed after the punched output is created.

<copies>

indicates the number of copies of the punched output file you want. One copy is the default.

User-Coded PCHOUT

A user-coded version of PCHOUT can include parameter entries for PC and PCP that resemble the = and asterisk (*) parameters of the PRINT command, which invokes the PBDPRN utility to print printer backup files. The MCP PCHOUT program ignores these parameters.

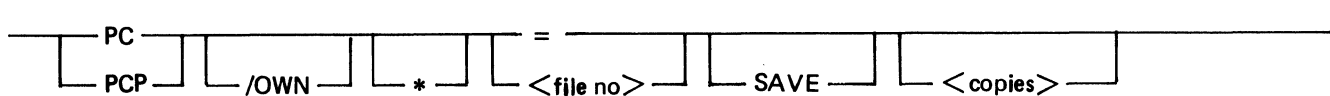


Figure 2-146. PC, PCP Command Syntax (User-Coded Version)

/OWN

Using the /OWN option prevents the scheduling of the bound MCP intrinsic when a user-provided PCHOUT is required but is not on disk.

If no program with this name is in the disk directory, the bound MCP intrinsic PCHOUT is scheduled.

=

indicates that all punch backup files on the specified medium will be punched.

*

informs the PCHOUT program that an AX response will provide further specifications, such as skipping lines or restarting, after operation begins.

After locating the specified file, PCHOUT displays "ENTER SPECS" and issues an ACCEPT. The response to the ACCEPT must be in one of the formats shown in Figures 2-167, 2-168, or 2-169 under the PRINT (Print Files) command.

Details

Backup files created by WFL, CANDE, RJE, and BNA cannot be punched by the PC command and cannot be accessed by programs that open "@00000" and "*00000". This feature prevents files from being accidentally removed before they can be printed (RJE), viewed (CANDE), printed by PRINT (WFL job), or routed back to a remote host (BNA).

Examples

PC /MCP 20010 SAVE

schedules the MCP bound intrinsic PCHOUT to create punched output of disk punch backup files number 20010. The punch backup file will be saved after the output has been created.

PCP /OWN 00104 SAVE 5; PRIORITY 5

schedules a user-coded version of PCHOUT to create punched output of a punch pack file with the number 00104 with a memory, processing, I/O, and schedule priority of 5. The backup file will be saved after the output has been created. Five copies will be punched.

See also:

CV, CVP (Convert Punch Backup File To Pseudo Card Deck); RO (Reset System Options); SO (Set System Options); and TO (Display System Option Status) commands.

The USE PCH, PCD, and PCP records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *V Series MCP/VS Programming Reference Manual* for information about PCHOUT.

PD (Print Directory)

Keyboard Input Message

Access Level: 0

The PD command lists the names of files on disk, pack, or LOADMP/PACKUP tape.

This command reads directories; it does not look for the actual files.

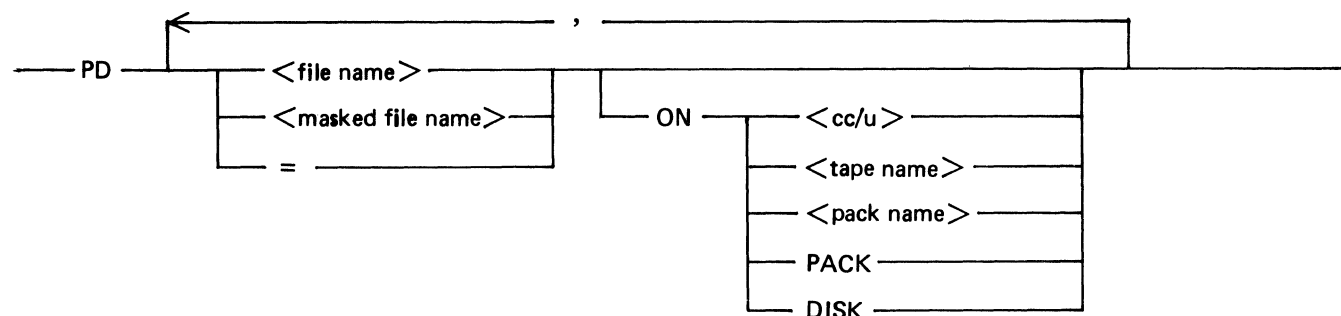


Figure 2-147. PD Command Syntax

<file name>

<masked file name>

=

indicates the name of the file. A single equal sign lists all files.

ON

If the ON clause is missing, or if the ON clause specifies DISK, the PD will be done on disk (100-byte media).

<cc/u>

lists files on a specific pack or tape drive. Both base and continuation areas are included.

Disk, Pack, and Tape PDs can be mixed in the same command. However, if no location is given, the default is disk.

<tape name>

specifies a tape which was not created by the SYSTEM/COPY utility. (Use the DIR (List File Directory) command.)

<pack name>

causes the base area of all packs with this name to be searched.

PACK

indicates all unrestricted packs.

DISK

indicates disk.

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

Examples

PD A, B, C

lists the files A, B, C contained in the disk directory.

PD = ON DISK

lists all files in the disk directory.

PD A= ON 3/12

lists all files beginning with A on channel number 3, unit number 12.

PD FILE01 ON SYSTEM

lists a file called FILE01 contained in the directory on a pack called SYSTEM.

See Also:

DIR (List File Directory) command.

PERFORM (Schedule DMPALL)

MCP Control Instruction

Access Level: 1

The PERFORM command schedules the DMPALL utility, which:

- Prints the contents of a disk, pack, magnetic tape, or card file.
- Reproduces a data file from one hardware type to another.

If there is a free-standing user-supplied program called DMPALL on disk, that program will be started. If not, the MCP-intrinsic DMPALL utility will be started.

Parameters are passed to DMPALL through a pass file on disk.

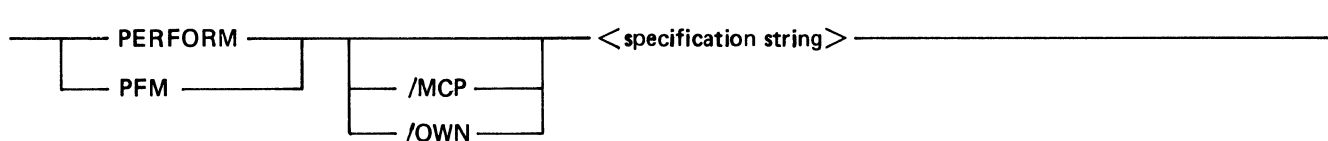


Figure 2-148. PERFORM Command Syntax

PERFORM

PFM

are synonymous.

/MCP

/OWN

specify whether the bound, MCP-intrinsic DMPALL or a user-coded DMPALL is required.

If /MCP is specified, the bound, MCP-intrinsic DMPALL is scheduled regardless of the presence of a user-provided DMPALL on disk.

The /OWN option can be used to prevent the scheduling of the bound, MCP-intrinsic DMPALL when a user-provided DMPALL is required and is not on disk.

<specification string>

See the PERFORM option of the DMPALL Intrinsic in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities* for the available options and the method of specifying their inclusion in the <specification string>.

Examples

PERFORM LST1 A DSK UA

produces a listing in UA format of a disk file named A.

PERFORM/MCP CRDCRD A B; CHARGE 99

schedules the MCP version of DMPALL to reproduce a card file named A to a new file named B and inserts a user charge number of 99 into the Run log for the task.

PFM/OWN DSKDSK D F 100 10 500 20; CHARGE 34

schedules a user-provided version of DMPALL to create a new file; changing the file id; specifying record size, blocking factor, records per area, and number of areas of a disk file; and inserting a user charge number of 34 into the Run log for the task.

See Also:

GENERATE (Schedule DMPALL) command.

The DMPALL utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities* and the *V Series Programmer's Reference Guide*.

PG (PURGE MAGNETIC TAPE)

Keyboard Input Message

Access Level: 7

The PG command purges a magnetic tape on a unit that is ready, has a write ring, and is not in use.

The purge operation retains the label type and physical tape number (if any) currently written on the tape. If the tape was previously unlabeled, a Unisys Standard label is written.

The syntax for the PG command is shown in Figure 2-149. However, the PG command also has other capabilities that let you assign a job to the tape after purging and/or override a program's tape type request. The syntax for these capabilities is shown in Figure 2-150.

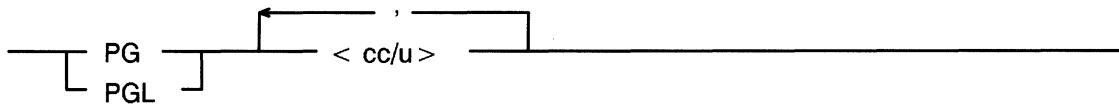


Figure 2-149. PG Command Syntax (Format 1)

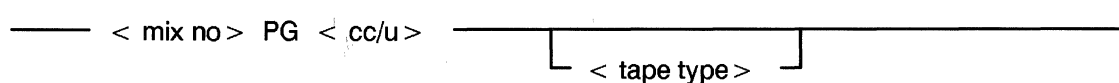


Figure 2-150. PG Command Syntax (Format 2)

Format 1

PG

purges the tape and leaves it ready and available to the system.

PGL

indicates the tape is to be locked after being purged.

<cc/u>

is the channel number and unit number where the magnetic tape is located.

Format 2

<mix no>

if specified and the selected job is waiting for an output tape, purges the tape and then assigns it to the job in the greatest tape density allowed by the program and the drive. You can specify <mix no> only in Format 2. Note that you cannot specify PGL and <mix no> together.

<tape type>

allows an override of a user program's tape type request only when you specify a <mix no>. The tape type can be MT9 (NRZ format), MPE, GCR or MTC.

Details

If you use format 1 (omitting <mix no> and <tape type>), and you use a GCR-capable drive, scratch tapes are written in GCR mode.

Examples

PGL 14/1, 14/2

purges and then locks the magnetic tapes on units numbers 1 and 2 on channel number 14.

41 PG 6/3 GCR

purges a GCR magnetic tape on channel number 6, unit number 3 and assigns it to the job with mix number 041.

See Also:

AC (Purge And Place Tape Number In Label); RP (Ready And Purge Magnetic Tape); RY (Ready Peripheral Device); and SN (Purge and Place Number In Tape Label) commands.

PM (PRINT MEMORY DUMP)

Keyboard Input Message

Access Level: 1

This command causes the analysis and printout of memory dump files by invoking one of two bound intrinsic programs: DMPOUT or DMPANL. DMPOUT analyzes a dump of specified user files; DMPANL analyzes a dump of the MCP file.

This command allows you to limit the scope of the printout for either program.

The general syntax for invoking these two programs is shown in Figure 2-151.

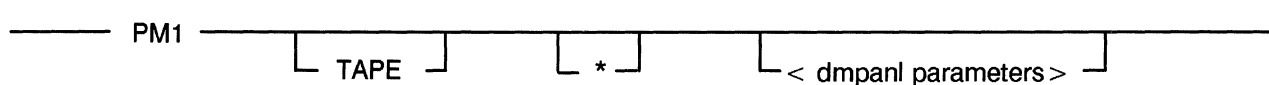


Figure 2-151. PM Command Syntax

<snnnn>

is the system number <s> of the system (0-3), and the dump file number <nnnn>.

To invoke DMPANL, you enter PM 1, or PM s0001, where s is the number of a system whose MCP file you want to analyze.

To invoke DMPOUT, you enter PM <snnnn>, where <nnnn> is the number of the user dump file and can range from 2-9999.

TAPE

indicates the source of the memory dump is on tape and is valid only for DMPANL.

* (multi-line input option)

allows you to enter multi-line input in the form of AX commands: <mix no> AX <parameter>.

For example:

5 AX FILE, DATAPAGE END

You must terminate the last AX command with the word END. If you enter <mix no> AX HELP, the system will display the parameters for the program (DMPOUT or DMPANL, depending on which you invoked).

<parameters>

allow you to limit the printout to specific tables and memory areas. The parameters for each program are described separately.

DMPOUT Parameters

The syntax for the DMPOUT parameters is given in Figure 2-152.

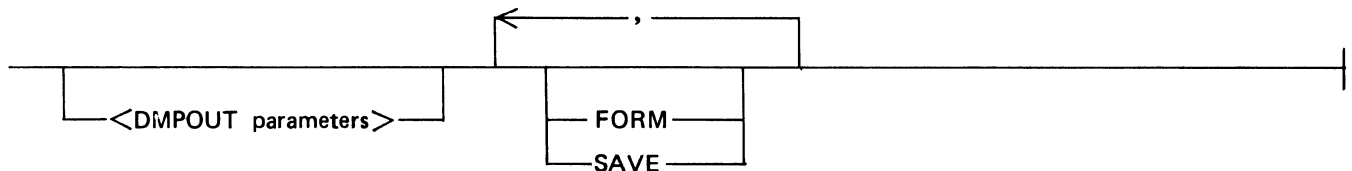


Figure 2-152. DMPOUT Command Syntax

FORM

allows you to send the listing to a particular printer by setting the special forms flag in the output printer file.

SAVE

keeps the dump file on disk after the analysis.

<DMPOUT parameters>

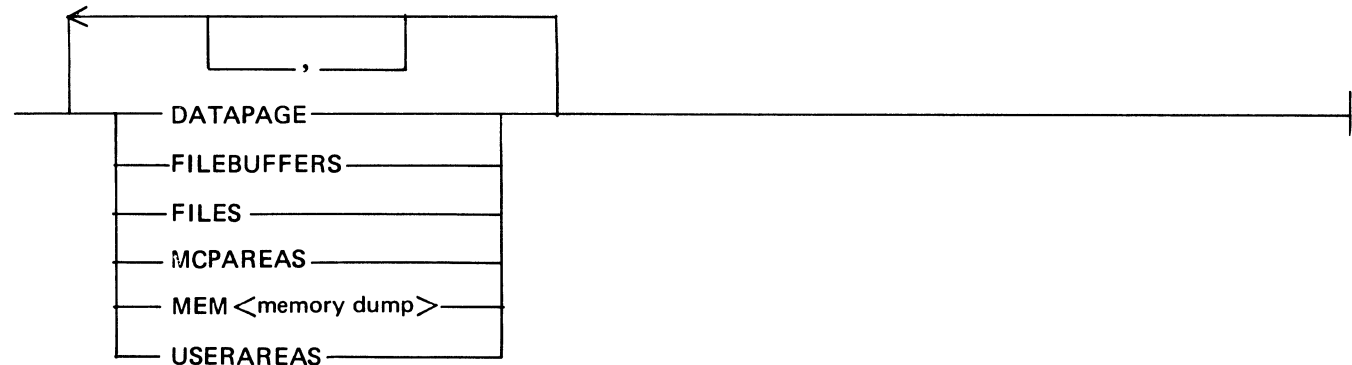


Figure 2-153. <DMPOUT parameters> Syntax

DATAPAGE

prints the task's MCP data page.

FILEBUFFERS

prints the same information as FILES, but also prints the file buffer of any file that has one.

FILES

prints the information about all of the open files and the files that were closed without release to be printed.

MCPAREAS

prints only the MCP data related to the task.

MEM

prints a raw memory dump of the area requested.

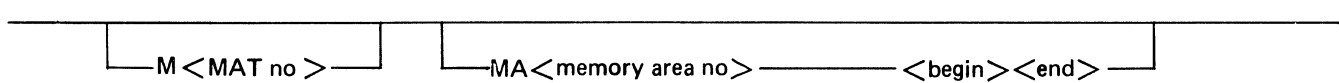


Figure 2-154. <memory dump> Syntax

M <MAT no>

defaults to the first MAT if no MAT is specified. For code files generated before MCP/VS 2.0, the only allowable MAT is the default.

MA <memory area no>

uses memory area one, the user memory area, if no memory area is specified.

<begin> – <end>

specifies the begin and the end addresses in KD. If they are not specified, the entire memory area is printed.

USERAREAS

prints only the user's data, code, and the basic information about the files that were open at the time of the dump. This is the default if no parameters are entered.

The DMPANL Parameters

The syntax for the DMPANL parameters is given in Figure 2-155.

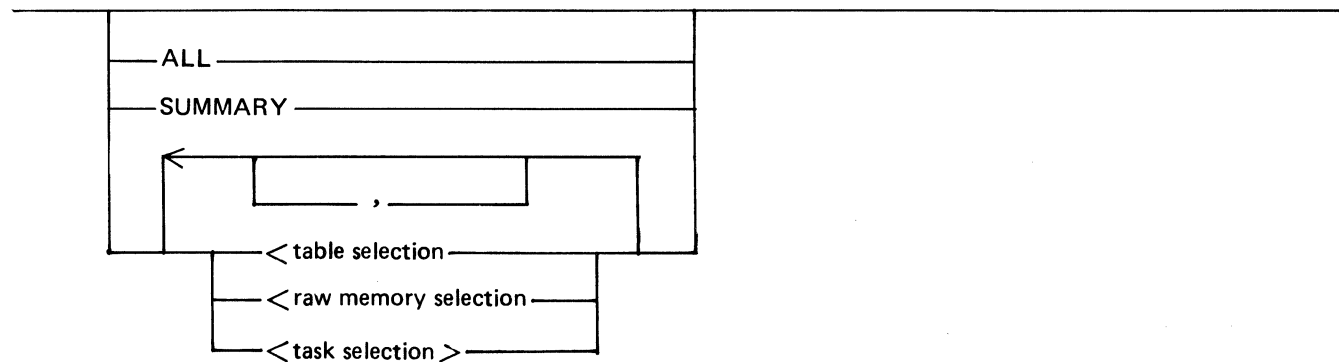


Figure 2-155. DMPANL Command Syntax

ALL

prints all of the loaded MCP, the system tables, and all other allocated memory.

SUMMARY

prints a brief summary of the dump, including the current task information, the function path, and the current task's MCP stack frames.

<table selection>

Table selections are shown in Figure 2-156.

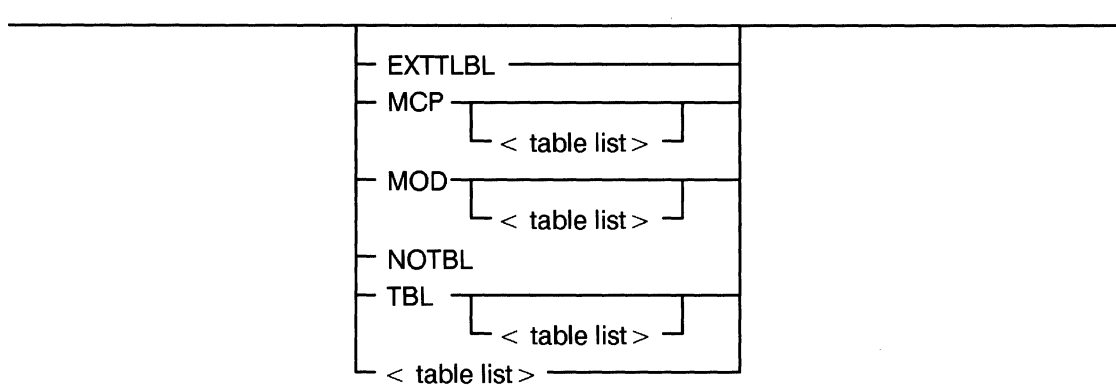


Figure 2-156. <table selection> Syntax

EXTTLBL

prints the system tables and all of the task MCP tables such as the soft IOATs, the memory resident DFHDRs, and the address blocks. No raw memory dump is produced.

MCP

has the same effect as not specifying any table selection, which is to produce a system table breakout and a raw dump of the MCP and overlay area.

MOD

produces a listing of the system tables, the memory from 0 KD to the top of the overlay area, any MCP modules that are currently loaded, and any Independent Runners that were executing.

NOTBL

overrides the default of printing the system tables when specific MCP tables are requested.

TBL

prints the system tables with no raw memory dump.

<table list>

- The DMPANL keywords for available system tables are listed in the following paragraphs.

- **Figure 2-157 deleted by PCN-001.**

CMPLX

prints information about tasks waiting in the complex wait function.

DCP

prints all of the information relating to the Data Communications Processors (DCPs). This includes the DCP Station Table, and the DCP and MCS tables, including their entries and buffers.

DMS

prints information in two DMSII data structures; the Database Program (DBP) Table and the Database User Program Control Table.

EU

prints the information in the EU table and in the disk subsystem table.

FILE

prints the information in the FIB, the IOAT, the external FIB, the memory resident DFHDR, the address blocks, and the Device Alternate Buffer for every file open on the system.

GLOBAL

prints the global code and overlay segment.

HLPARMS

prints the halt/load parameters.

IOAT

outputs the information in the IOAT table.

IOQUE

prints the information about each channel and about the queue elements that are in process and waiting to be fired.

- JMR

- prints all internal data structures used by the Job Manager.

- MAST

prints the Memory Area Status Table.

MAT

prints the Memory Area Tables for every task and MCP function.

MIX

prints the Mix Table.

MSG

outputs the messages currently waiting in the Message Module's message pool.

PORTS

prints the information in the Port Global Data Memory Area and the Candidate List Memory Area.

The Port Global Data Memory Area contains an entry for every active PORT on the system. Each entry contains attributes that are shared by all the Subports connected to this PORT, including an event directory.

The Candidate List Memory Area contains an entry for every Subport that is trying to open. An entry contains attributes necessary for matching Subports and control state and status information.

SEC

prints the security user table.

SEG

prints the segment dictionary table.

SLOG

prints the last 100 records of the ODT log.

SNAP

prints a formatted listing of the SNAP picture. SNAP pictures are inserted into memory under certain program or system failure conditions. The contents of a SNAP picture differs depending on the type of V series system (V 300, V 500, and so forth).

SUBPORTIO

prints the information in the Shared Subport Memory Area (SSMA). There is one SSMA for each pair of open Subports on the system. An area contains the input and output data queues and attributes shared by both connected Subports.

TRAK

prints the TRAK buffer.

TS

prints the Time Sharing Processor Information Block and the Shared Area Table.

VCS

prints internal data structures used by the V Series Communication System (VCS).

WAIT

prints the information about each task waiting on the system, including the locks each owns and the locks each is attempting to get.

<raw memory selection>

prints memory from the specified beginning address to the specified end address. No system tables are printed.

—— RAW < beginning> < end> ————|

Figure 2-158. <raw memory selection> Syntax

<task selection>

prints information about the task whose number is specified. The default printout contains all of the task's memory areas and the MCP data relating to the task. You can limit the printout by specifying any of the task selection parameters.

———| TASK < task no> ————|
 | |
 | < task parameters> |

Figure 2-159. <task selection> Syntax

<task parameters>
indicates the task parameters as shown in Figure 2-160.

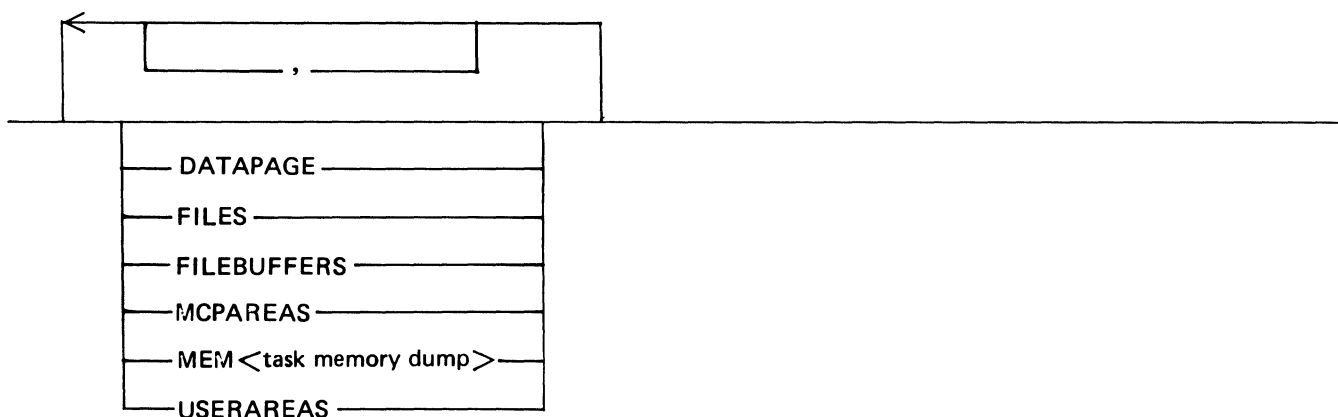


Figure 2-160. <task parameters> Syntax

DATAPAGE
prints the task's MCP data page.

FILES
outputs the information about all of the files that are opened, or that were opened and closed without release.

FILEBUFFERS
prints the same information as FILES, and also prints the external file buffers of any files that have them.

MCPAREAS
prints only the MCP data related to that task.

MEM
prints a raw memory dump of the area requested.

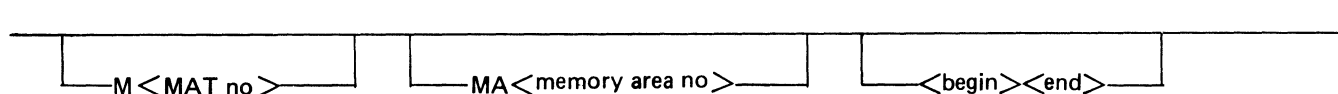


Figure 2-161. <memory dump> Syntax

M <MAT no>
defaults to the first MAT if no MAT is specified. For code files generated before MCP/VS2.0, the only allowable MAT is the default.

MA <memory area no>

uses memory area one, the user memory area, if no memory area is specified.

<begin>

<end>

specifies the begin and the end addresses in KD. If they are not specified, the entire memory area is printed.

USERAREAS:

outputs only the user's data and code.

Examples

PM 1 TBL

prints the system tables with no raw memory dump.

PM 1 DCP MIX

prints all information relating to DCPs, as well as the Mix Table.

PM 23 DATAPAGE SAVE

prints the MCP data page for the user dump file 23, and leaves the dump file on disk.

See Also:

The DMPANL utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

Discussions of memory in the *V Series MCP/VS Architecture Manual*.

PO (Power Off Pack)

Keyboard Input Message

Access Level: 7

The PO command powers off the designated pack unit once all active files resident on the pack have been closed.

PO <cc/u>

Figure 2-162. PO Command Syntax

<cc/u>

is the channel number/unit number location of the pack.

Restrictions

This command is not valid for MD4, MD8, or 680 pack type.

The drive:

- cannot be SHARED
- cannot have any areas in use or be assigned to any task
- cannot be in use by direct I/O
- must have been previously SAVED
- must be ready. A pack can be made ready through the RY (Ready Peripheral Device) command.

Example

PO 19/1

powers off the pack located at channel number 19, unit 1.

See Also:

RY (Ready Peripheral Device) and SV (Save A Peripheral Unit) commands.

PP (Print Pack Sectors)

Keyboard Input Message

Access Level: 5

PP invokes the intrinsic program DSKOUT to print selected sectors from a pack. The starting address and numbers of sectors given at execution time specify which sectors to print.

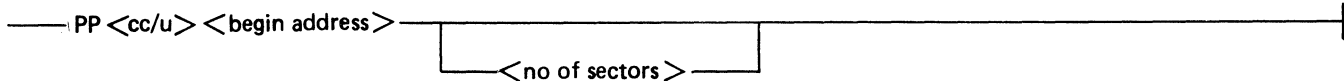


Figure 2-163. PP Command Syntax

<cc/u>

is the channel number/unit number location of the pack.

<begin address>

specifies the beginning point for the printing. The limits are 0 to 16700000 (or the pack limit) and must be specified in decimal format.

<no of sectors>

indicates the number of sectors to be printed. If this element is omitted, a value of 1 is assumed. If specified, the value must be less than 10000000.

Output Options

Various output options to provide alternate sorting and analysis are available. A full description of these options and their applications appear under the KA (Analyze Disk Directory) command.

Examples

PP 10/11 2500 10

prints 10 sectors from the pack located at channel number 10, unit number 11, with the starting address of 2500.

PP 1/2 6

prints 1 sector from the pack located at channel number 1, unit number 2, with the starting address of 6.

See Also:

KA (Analyze Disk Directory); KP (Print Disk Segments); KS (Analyze Disk Space); PA (Analyze Pack Directory); and PS (Analyze Pack Space) commands.

PR (Change Priority)

PRM (Change Memory Priority)

PRP (Change Processing And I/O Priority)

PRS (Change Scheduling Priority)

Keyboard Input Message

Access Level: 3

The PR command changes the memory, processing and I/O, and schedule priorities of a program in the mix or schedule. You can change all the priorities to a common setting at the same time or change each individually.

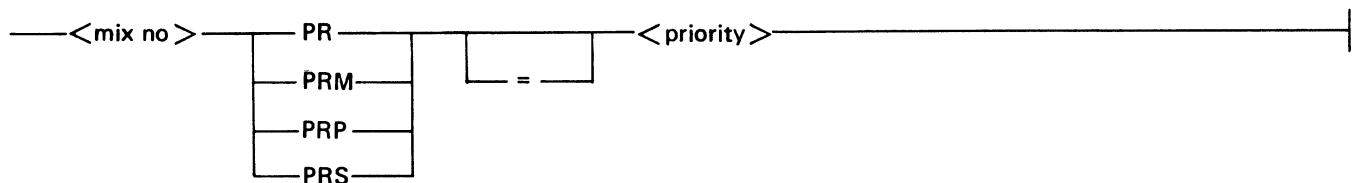


Figure 2-164. PR, PRM, PRP, PRS Command Syntax

<mix no>

is the mix number of the program whose priority is to be changed.

PR

changes and provides a common priority setting for the memory, processing and I/O, and schedule priority of a program.

PRM

changes the memory priority of the specified program.

PRP

changes the processing and I/O priority of the specified program.

PRS

changes the schedule priority of the specified program. The schedule priority can be changed only while the program is in the schedule and not yet running.

=

is optional and does not influence the command.

<priority>

is a number ranging from 1 to 9, with 1 the lowest priority and 9 the highest. The MCP assigns a default priority of 4 unless directed otherwise by certain control commands and defaults.

Examples

2 PR 5

causes the program whose mix number is 2 to have a memory, processing and I/O, and schedule priority of 5.

4 PRM 3

causes the program whose mix number is 4 to a memory priority of 3.

3 PRP 9

causes the program whose mix number is 3 to have a processing priority of 9, the highest priority.

2 PRS = 6

causes the program whose mix number is 2 to have a schedule priority of 6.

See Also:

PRIORITY (Set Common Priority) command.

The CONTROL PRIORITY record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

PRINT (PRINT FILES)

MCP Control Instruction

Access Level: 1

PRINT invokes the PBDPRN system utility that prints files on various printing devices, including the Unisys Image Page Printer (IPP).

PBDPRN can do the following:

- print printer backup files by number or name
- print printer backup files from tape
- create printer backup tapes
- search the backup file for a specified string of text and begin printing when it finds it
- start printing at a specified position in a file and stop at another position
- translate all lowercase letters to uppercase letters as it prints
- skip lines for the printout
- stop printing after a specified number of lines
- restart the printing of print job that had been stopped by the QT (Quit Program Operation) command

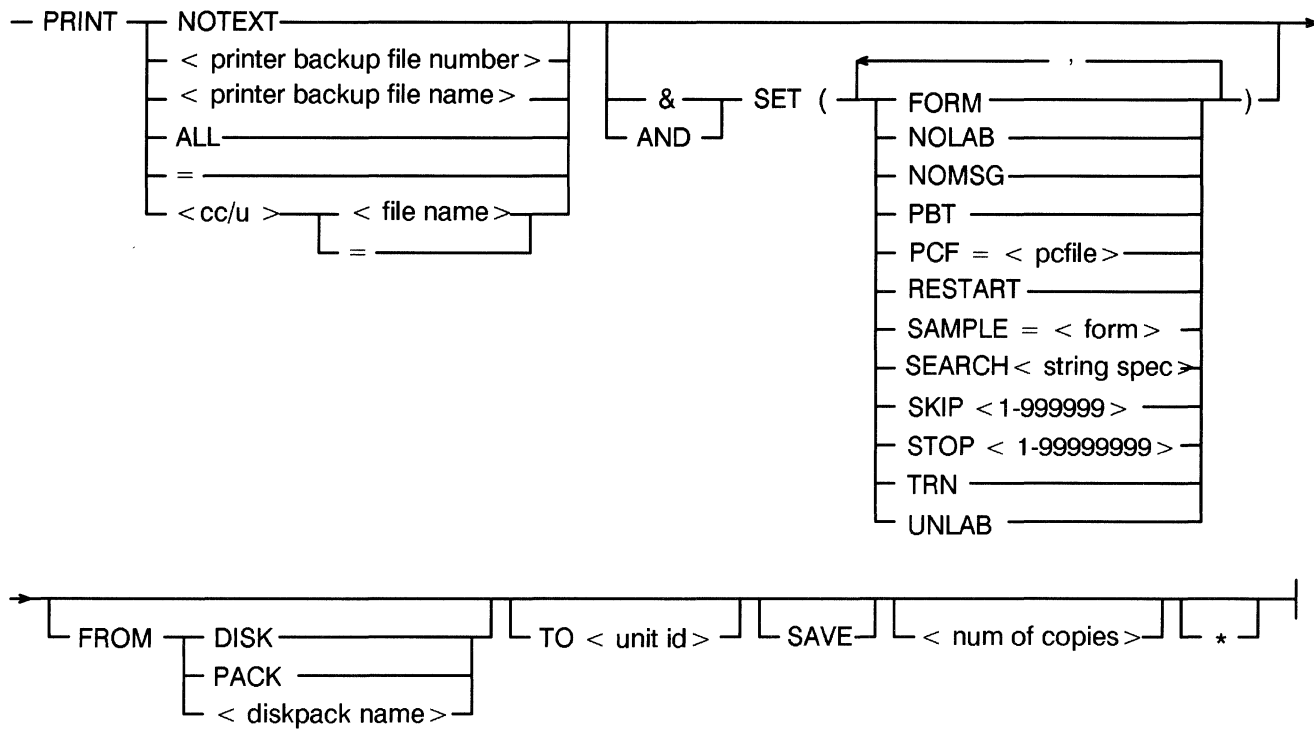


Figure 2-165. PRINT Command Syntax

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

NOTEXT

prints a sample of a form or a printer control file (PCF) produced by the Forms Manager program. This option is used only with the IPP.

<printer backup file no>

<printer backup file name>

prints individual files according to their backup file number or backup file name.

ALL

=

causes PBDPRN to print all printer backup files. Unless DISK or PACK is specified, this option prints all the printer backup files on both disk and pack.

<cc/u>

<cc/u> <file name>

<cc/u> =

PBDPRN can print printer backup files from tape if you specify the tape drive's channel number/unit number. <cc/u> by itself prints all printer backup files from the specified tape. If a file name is used, the system prints only that file. If the equal sign is used, the system prints all the printer backup files on that tape.

AND SET

& SET

are synonymous. The options following AND SET (& SET) must be within parentheses. If you are using more than one option, separate each with a comma.

FORM

directs output to the printer previously specified in the MCP file equate control card. For example, FILE PRINT = SHORT/OUTPUT.

NOLAB

stops a block letter label from being printed on the cover page of the printout. However, the line that indicates the backup file number, its name, and the file label information will be printed.

NOMSG

stops PBDPRN from displaying messages on the system console when it finishes printing copies of a file.

PBT

directs the printer backup listing to tape.

PCF = <file name>

must be used to print backup files on an Image Page Printer. When NOTEXT is specified, an entire printer control file (PCF), which can contain multiple forms, is printed as a sample.

RESTART

restarts a print job that has been stopped with a QT command or as a result of some IPP VSID message protocol errors that terminated PBDPRN.

This option tells PBDPRN to start printing at the beginning of the page it was printing when it was stopped. If it does not find the beginning of a page, it resumes at the beginning of the last line it was printing when it was stopped. This option does not work for files on printer backup tapes.

For this option, use the same syntax used to start the job originally, but include the option RESTART.

SAMPLE = <form>

prints sample forms in the process of being created by the Forms Manager program. It prints only the form, no text. It is used only with an IPP and is reserved for use by the Forms Manager Program.

SEARCH <string spec>

The following diagram illustrates the syntax for the string specifications.

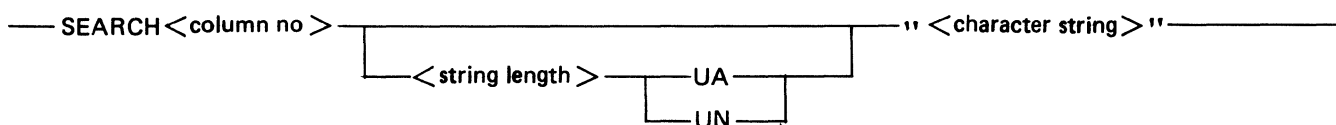


Figure 2-166. SEARCH <string spec> Syntax

SEARCH

tells PBDPRN to search a column of text and to start printing the file where it finds a matching string of characters. The string cannot exceed 99 characters.

<column no>

tells PBDPRN which column to search. It can be any number between 1 and 132. Note that the sum of the column number and the string length cannot exceed 133.

"<character string>"

causes PBDPRN to search for this string of characters. The string cannot be longer than 99 characters, including blanks.

<string length>

indicates the number of characters and blanks in the string. There can be up to 99 characters in the string.

UA

UN

represent data types. UA indicates the string is a representation of bytes as seen on a printed listing.

SKIP <1-999999>

tells PBDPRN to begin printing at a certain line number and can be used with the option STOP to print only part of a file.

STOP <1-99999999>

tells PBDPRN to stop printing at a certain line number and can be used with the option SKIP to print only part of a file.

TRN

tells PBDPRN to translate all lowercase letters to uppercase letters.

UNLAB

causes no label or other types of information to be printed on the cover page of the printout.

FROM DISK

FROM PACK

FROM <diskpack name>

FROM DISK indicates the printer backup file is on disk.

FROM PACK indicates the file is on diskpack. PBDPRN searches 1) the primary backup diskpack, 2) the secondary backup diskpack (if defined) and 3) all unrestricted diskpacks. Primary and secondary backup diskpacks are declared during system initialization and can be displayed or changed using the SHOW or ALTER commands.

FROM <diskpack name> indicates the printer backup file is stored on a specific diskpack. If no FROM clause is included in the PRINT command, PBDPRN searches both disk and pack by default.

TO <unit ID>

allows you to use a particular printer. If you are using an IPP, you must use the option PCF = <file name>. If this option is not used, PBDPRN prints the file on the first available line printer.

SAVE

stops PBDPRN from removing the printer backup file after it is printed.

<no of copies>

allows you to print more than one copy. Up to 99 copies can be printed for each print job. This option is ignored if PBDPRN is printing backup files from a tape.

*

tells PBDPRN to go into a waiting accept condition. PBDPRN will display ENTER SPECS on the ODT and prompt you with an ACCEPT. You can then AX parameters to PBDPRN with the MCP system command AX. The use and syntax of the parameters appears in Figures 2-167, 2-168 and 2-169.

—— < mix no > AX < skip count > ————

Figure 2-167. AX Response Syntax (Option 1)

Option 1 specifies the number of records to skip before beginning printing.

<mix no>

refers to the mix number of the job you want to respond to.

<skip count>

must be a one- to six-digit unsigned integer.

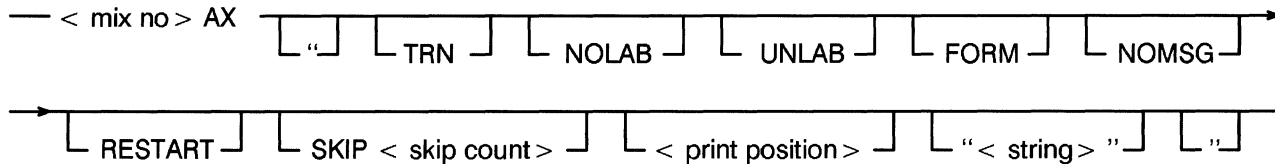


Figure 2-168. AX Response Syntax (Option 2)

All the elements in Option 2 are optional and can appear in any order except the <print position>-<string> option, which must be the last entry, if used.

TRN

translates all lowercase alphabetic characters to uppercase.

NOLAB

suppresses the printing of block-printed labels. The first line (specifying file-name, and so forth) still appears unless the file name is absent or labels are omitted.

UNLAB

suppresses printing of labels completely. Equivalent to the effect of declaring "LABELS OMITTED" on the backup file.

FORM

forces the output to be declared special forms.

NOMSG

PBDPRN normally displays a message on the ODT indicating that it has completed printing a copy of the file when multiple copies are requested. This option suppresses these messages.

RESTART

can be used to resume printing of a file that was previously ended with a QT (Quit Program Operation) command. When this option is specified, PBDPRN searches backward from the last line printed, looking for a line containing a "SKIP TO CHANNEL ONE" code. If such a line can be located within 100 lines of where the file was QTed, printing is resumed immediately following this line; if not, printing resumes with the line following the last one printed when the file was QTed.

SKIP <skip count>

performs the same function as in Option 1. If multiple SKIPS are present, only the last is used. However, all SKIPS are syntax-checked for being within the file limits.

STOP <stop count>

terminates the printing of the file at the specified record number. If SAVE is not specified on the PC command, the file is removed at the completion of printing. This option can be used in conjunction with the SKIP option to print only part of a file.

<print position>--<string>

specifies that printing is to begin when the specified string is found at the specified print position. The print position is a one- to three-digit unsigned integer, less than 133, specifying the column position in which the string starts. The first print position is one. The length of the string added to the print position cannot exceed 133. The string must be enclosed in quotes and it must follow COBOL rules. For example, an embedded quote must be represented by two consecutive quotes. "ISN""T" represents the string ISN"T, which is five characters long.

<mix no>AX<print position><length>UA<string>

Figure 2-169. AX Response Syntax (Option 3)

Option 3 is the format for search-string entries. <mix no> and <print position> perform the same functions as in Option 2.

<length>

is an integer specifying the length of the string to be searched.

UA

indicates that an alphanumeric compare will be performed.

<string>

is the actual data to be searched. It is not enclosed in quotes and begins at the first non-blank character following UA and continues to the end of the accept message. It must be exactly the same length as specified by the length parameter. If the string contains any special characters, the entire ACCEPT message must be enclosed in quotes.

Examples

PRINT 12345

prints the printer backup file @12345 on any available line printer.

PRINT 12345 AND SET (SKIP 100, STOP 1000)

prints the printer backup file @12345 starting at line 100, and stops printing at line 1000. Because no printer was specified, the file will be printed on any available line printer.

PRINT 6/1 =

prints all the backup files from the tape on channel number 6, unit number 1. Because no printer was specified, the file will be printed on any available line printer.

PRINT BKFILE & SET (SEARCH 10 "SAM") TO SPEC 5

searches columns 10-12 of each line and prints from the first occurrence of the string SAM to the end of the file. The system will use the printer SPEC and print five copies.

PRINT 12345 AND SET (PCF=DOCPCF)

prints the file @12345 on the IPP using DOCPCF form file.

PRINT NOTEXT & SET (PCF=FORMSS)

prints a sample of a form created by the Forms Manager program, printing only the form, no text.

NOTE

The DOCPCF and MCPPCF are two printer control files available for general use on the image printer. DOCPCF will format printer backup data in portrait mode (80 characters by 60 lines). MCPPCF will format printer backup data in landscape mode (132 characters by 60 lines), as well as in duplex mode. To create other forms and PCFs, the Forms Manager program can be employed.

Executing PBDPRN

PBDPRN can be executed with the MCP system command EX by using the following syntax: EX PBDPRN

After it is executed, PBDPRN sends the following message:

PBDPRN=<mix no> ENTER SYNTAX. AX 'END' TO TERMINATE INPUT.

Auto Printing

An MCP file equation command can be used to direct printer backup files to the Image Page Printer (IPP) automatically.

The following paragraphs describe the different syntax used to output files automatically to a line printer versus an IPP.

?FILE LINE = SPEC/OUTPUT FORM AUTO.

A sample command to output this file to an IPP:

?FILE LINE = DOCPCF/OUTPUT FORM AUTO.

NOTE

DOCPCF is the unit id for an IPP.

PBDPRN System Software

The B9290-30 IPP utilizes a high-level software protocol, known as Virtual Static Image Device (VSID), that communicates with the IPP. The PBDPRN utility uses the VSID protocol to communicate with the IPP unit.

The system software that runs and manages the IPP unit consists of a Resident Operating System (ROS), Loadable Operating System (LOS), and the Loadable Diagnostics. The ROS is a PROM-based module that resides on the GASP card in the IPP unit. The LOS is a software module that is downloaded from the host system. The Loadable Diagnostics is the software that is used to inform the user of system errors regarding the IPP.

The following commands can be entered at the ODT to download the software necessary for IPP operation.

1. Set the IPP option on the system with the ODT command: SO IPP
2. Do an "OL IPP" to check whether an IPP has been added as a peripheral device. If not, then enter:

UNIT <cc/u> MCPPCF IPP

where MCPPCF is the unit id recognized by PBDPRN and also found in the PRINT command syntax diagram; and where IPP is the unit id recognized by the system.

3. To download the LOS, the maintenance panel on the IPP console should first display: "RESIDENT OPERATING SYSTEM V <version #>". Enter: "LH <cc>", where cc is the channel number of the IPP unit. When the Load Host is complete, the maintenance panel on the IPP console should display:

"LOADABLE OPERATING SYSTEM V <version #>

4. The IPP unit is now ready to print.

See also:

The PBDPRN utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

PRIORITY (Set Priority)

PRM (Set Memory Priority)

PRP (Set Processing And I/O Priority)

PRS (Set Schedule Priority)

MCP Control Instruction

Access Level: 3

The PRIORITY command assigns a common setting of the memory, processing, I/O, and schedule priorities for a program at the time of scheduling. To set the different types of priority individually, use the PRM, PRP, and PRS commands.

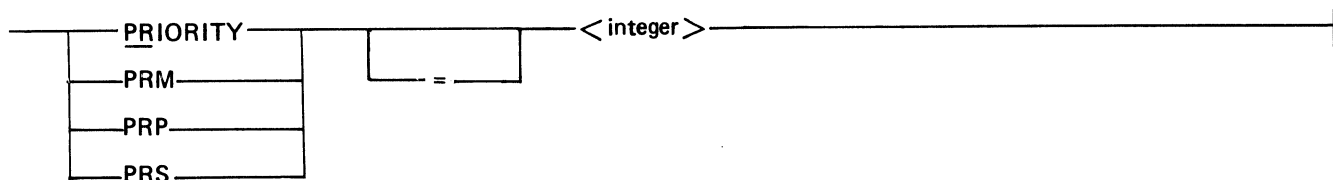


Figure 2-170. PRIORITY, PRM, PRP, PRS Command Syntax

PRIORITY

PR

are synonymous. Do not confuse PR with the PR (Set Priority) command, which is a keyboard input message that changes the priority of a job already in the mix or schedule.

PRM

specifies the memory priority to be assigned to a given program.

PRP

specifies the processing and I/O priority to be assigned to a given program.

PRS

specifies the schedule priority to be assigned to a given program.

=

is optional and does not influence the command.

<integer>

denotes the priority to be assigned. Priorities can range from 1 to 9, with 1 the lowest priority and 9 the highest. The MCP assigns a default PRIORITY of 4 unless directed otherwise by certain control commands and defaults.

Details

PRM

The PRM command indicates the priority to be given a program when allocating memory space. The PRM command applies only after the MCP determines that a program's schedule priority qualifies it for the mix.

If a program has sufficient schedule priority to enter the mix and sufficient memory is not available, the memory priorities of all jobs in the mix are compared with the memory priority of the job in the schedule.

If any jobs in the mix have a memory priority less than the memory priority of the job in the schedule, they can be rolled out from memory to disk, creating space for the higher-memory-priority job. These programs are then marked as PRIORITY STOPPED and take precedence over jobs of equal or lower memory priority when memory becomes available.

In addition, if a job in the mix requires memory for any reason (such as, disk file header space, or printer backup space), jobs with a lower memory priority can be priority-stopped to make memory available.

PRP

The MCP uses the PRP priority to determine the processing reinstate priority to be given to a program while in the mix. This priority does not affect job entrance into the mix from the schedule. While jobs are in the mix, however, the PRP priority establishes the order of reinstatement when the jobs are ready to run.

NOTE

A processor-bound program with a high process priority can prevent all jobs with a lower priority from being reinstated by the MCP.

PRS

The PRS command allows you to assign a schedule priority to a program. This schedule priority establishes which jobs receive precedence to enter the mix. The program is not affected after it has entered the mix.

No job enters the mix, regardless of PRP and PRM priorities or memory availability, until all jobs of higher schedule priority have been initiated. Jobs with equal schedule priorities are initiated (assuming memory is available) based first on memory priority, then length of time in the schedule.

Examples

COMPILE B PASCAL; PRIORITY = 5

compiles the Pascal program called B and assigns it a priority setting of 5.

EX C; PRM 7

executes a program called C and assigns it a memory priority setting of 7.

See Also:

PR (Change Priority) command.

The CONTROL PRIORITY record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

PS (ANALYZE PACK SPACE)

Keyboard Input Message

Access Level: 5

PS starts the intrinsic program DSKOUT to perform a detailed analysis of the space allocation on the designated pack.

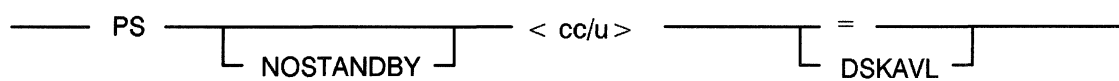


Figure 2-171. PS Command Syntax

■ NOSTANDBY

directs the operating system to capture information for DSKOUT even when the operating system *cannot* perform its normal standby function. The standby function is still attempted, but information is captured regardless of whether the function is successful. For more information about the standby function, see Details on the following pages.

<cc/u>

l number/unit number location of the pack.

=

causes analysis of the entire pack.

DSKAVL

causes an analysis of the available space on the pack only.

Details

Files, available areas, missing, and duplicate pack are all listed. Each file's area addresses, file identifier, area size, area number, and file attributes are given.

For available areas, the area address, the string "A V A I L A B L E", and the size of the available space are given.

For missing or duplicate pack, the address of the area, the string "MISSING PACK" or "DUPLICATE PACK", and size of the area are given.

- When you enter a PS command, the operating system can respond with the message:

- KBD IGNORED: DISK OR PACK MAINT IS IN PROGRESS

This means the operating system cannot perform a standby function as part of the PS command. A standby function suspends updates to operating system data structures long enough to obtain an accurate "snapshot" of the structures. This "snapshot" is passed to DSKOUT for analysis. The operating system needs exclusive access to these structures for an accurate DSKOUT analysis.

To respond to this situation, you can:

- Wait until a later time when pack activity is less
- Use the NOSTANDBY option of the PS command

NOSTANDBY captures information for DSKOUT even when the operating system *cannot* perform the standby function. The standby function is attempted, but information is captured regardless of whether it is successful.

Using NOSTANDBY can create missing or duplicate pack messages that are not correct. This happens if an update occurs as the “snapshot” is taken. If NOSTANDBY is used and missing/duplicate pack conditions are encountered, DSKOUT flags the resulting messages with NOSTANDBY warnings.

These warnings mean the condition may be caused by non-exclusive access rather than actual missing or duplicate pack. Before you act on any missing/duplicate pack messages with a NOSTANDBY warning, enter another PS command without NOSTANDBY. Enter the command when less pack activity is taking place.

Output Options

Various output options to provide alternate sorting and analysis are available. A full description of these options and their application appear under the KA (Analyze Disk Directory) command.

Examples

PS 10/1 =

invokes the DSKOUT intrinsic to analyze only the space allocation on the pack on channel 10, unit number 1.

PS 19/3 DSKAVL; CHARGE 6610

invokes the DSKOUT intrinsic to analyze only the available space on the pack on channel 19, unit number 3. This job will be assigned a charge number of 6610.

PS NOSTANDBY 14/0 =

invokes DSKOUT to analyze only the space allocation on the pack on channel 14, unit 0. The analysis is generated regardless of the outcome of the normal standby function.

See Also:

KA (Analyze Disk Directory); KP (Print Disk Segments); KS (Analyze Disk Space); KX (Obtain User Disk Space); PA (Analyze Pack Directory); and PP (Print Pack Sectors) commands.



QD (Query DEBUG Sessions)

Keyboard Input Message

Access Level: 8

The QD command identifies the DEBUG sessions present on the system and displays the four-digit session number of each.

— QD —

Figure 2-172. QD Command Syntax

See Also:

DEBUG (Attach Program To DEBUG Session), ED (Enter DEBUG Session), and ID (Invoke DEBUG Session) commands.

The DEBUG utility in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

QT (Quit Program Operation)

Keyboard Input Message

Access Level: 1

The QT command can do the following:

- stop the printing or punching of a backup file
- interrupt the DMPALL or DMPANL utility and cause it to go to a normal EOJ
- terminate a timesharing program, so you do not have to discontinue it with a DS (Discontinue Program) command
- terminate a Work Flow Language (WFL) job that has the RESTART option enabled

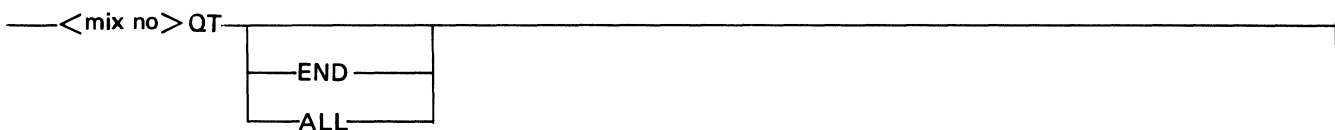


Figure 2-173. QT Command Syntax

<mix no>

is the mix number of the program to be stopped.

You can use QT only on the DMPALL, DMPANL, DMPOUT, DSKOUT, LDCNTL, PBDPRN, and PCHOUT utility programs; jobs from the CANDE shares mix; and any restartable WFL job.

NOTE

You cannot use the QT command on a locked program. If you want to use QT on a restartable WFL job, you must first unlock the job with the UP (Unlock Program) command.

ALL

causes the designated intrinsic to immediately stop printing or punching all files, leaving all files not previously purged on disk or pack.

ALL terminates timesharing if no users are present and does not allow further log-ins.

END

causes the specified intrinsic to stop printing or punching at the end of the current file and saves or purges the files as requested in the PRINT (Invoke PBDPRN), PC (Punch Backup File), or PM (Print Memory Dump) command.

END terminates timesharing if no users are present in shared areas only.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Details

If you omit the optional END and ALL specifications, the print or punch utilities immediately stop printing or punching the current file. The utilities go to the next file, if one has been requested, and leave the previous file on disk.

If you are quitting a timesharing program and omit the END or ALL option, the timesharing process terminates immediately.

If the QT command is used to terminate a restartable WFL job, the WFL job will remove its own entry from the restart file and then terminate itself. It cannot then be inadvertently restarted by a subsequent EXECUTE (Execute Library Program) command.

Examples

3 QT

stops processing the program with mix number 003.

5 QT ALL

stops the program with mix number 005 from printing or punching all files, leaving all files not previously purged on disk. If the program is a CANDE job and no users are present, timesharing will be ended and no users will be able to log in.

See also:

DS (Discontinue Program); PC (Punch Backup File); PRINT (Invoke PBDPRN); PM (Print Memory Dump); and UP (Unlock Program) commands.

The *V Series CANDE Installation and Operation Guide*.

The *B 2000/B 3000/B 4000/V Series WFL Compiler Programming Reference Manual*.

QWIKPOOL (Set Program Overlay Buffer Pool)

MCP Control Instruction

Access Level: 8

QWIKPOOL sets the total size of the user program overlay buffer pool. The pool is used to store program overlays.

QWIKPOOL overrides the buffer size set at cold start by the system default or by the LIMIT QWIKPOOL record.

You can enter the QWIKPOOL command only when the QWIK option is reset. See RO (Reset System Options) for information about resetting the QWIK option.

— QWIKPOOL <integer> —

Figure 2-174. QWIKPOOL Command Syntax

<integer>

specifies the pool size in thousands of digits (KD) and can have any value from 1 KD to 9999 KD. The default setting is 400 KD.

Details

The user QWIK pool is memory dedicated to storing the most recently used user program overlays, so that they will not have to be accessed from disk if they are needed again. This feature usually increases performance. You can limit the amount of memory to be used by the QWIK pool with the LIMIT QWIKPOOL record of the system configuration file. You can set the QWIK pool option with the USE QWIK record or by the SO (Set System Options) command. It can be reset by the RO (Reset System Options) command.

User QWIK memory differs from MCP QWIK memory in that MCP QWIK memory is memory dedicated to storage of the most recently used MCP overlays and is set by the LIMIT QWKMEMP record in the system configuration file.

Example

```
RO QWIK
QWIKPOOL 600
SO QWIK
```

resets the QWIK system option and then sets total size of the user program memory buffer pool to 600 KD. Then the QWIK option is reenabled.

See Also:

QWKMEMP (Set MCP Overlay Buffer Pool); RO (Reset System Options); SO (Set System Options); and WO (Display Active User QWIK Operation Counts) commands.

The LIMIT QWIKPOOL and USE QWIK records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

QWKMEM (Set MCP Overlay Buffer Pool)

MCP Control Instruction

Access Level: 8

The QWKMEM command allocates a fixed amount of memory exclusively for MCP overlay use.

— QWKMEM <integer> —

Figure 2-175. QWKMEM Command Syntax

<integer>

specifies the amount of memory in thousands of digits (KD) to be reserved for MCP use and can have any value from 0 KD to 999 KD.

Details

MCP QWIK memory differs from user QWIK memory in that MCP QWIK memory is memory dedicated to storage of the most recently used MCP overlays and is set by the LIMIT QWKMEM record in the system configuration file.

Example

QWKMEM 60
designates 60 KD of memory for MCP use only.

See Also:

QWIKPOOL (Set Program Overlay Buffer Pool) command.

The LIMIT QWKMEM record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

RA (Remove After-Linkage)

Keyboard Input Message

Access Level: 1

The RA command removes after-linkage for the indicated task, allowing it to enter the mix.

— <mix no> RA —

Figure 2-176. RA Command Syntax

<mix no>

must be the mix number of a task that is in the schedule and cannot enter the mix because it is to be executed after another task.

Details

An after-linkage is a state that exists between two tasks when the AFTER (Hold Program) command is used.

For example:

```
EX TASK1
EX TASK2 AFTER TASK1
```

First, TASK1 is executed. Then, an AFTER command is issued that specifies TASK2 is to wait in the schedule until TASK1 finishes executing. An after-linkage now exists between the two tasks.

If TASK1 does not go to a normal EOJ and the AFTR option is reset (off), you can do one of the following:

- use the RA command to remove the after-linkage if you want TASK2 to enter mix.
- use the RS (Remove Job From Schedule) command to remove TASK2 from the schedule.

Example

27 RA

removes the after-linkage of a task in the schedule with mix number 027.

See Also:

AFTER (Hold Program); DS (Discontinue Program); and RS (Remove Job From Schedule) commands.

The USE AFTR record in system configuration file in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

RD, RDP (Remove Pseudo Decks)

Keyboard Input Message

Access Level: 5

RD removes pseudo card decks from the system. However, currently active pseudo decks are not affected.

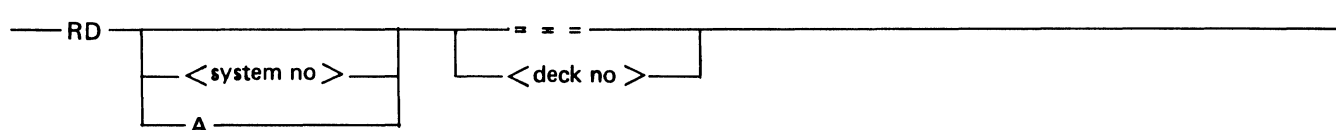


Figure 2-177A. RD, RDP Command Syntax (Format 1 - Disk)

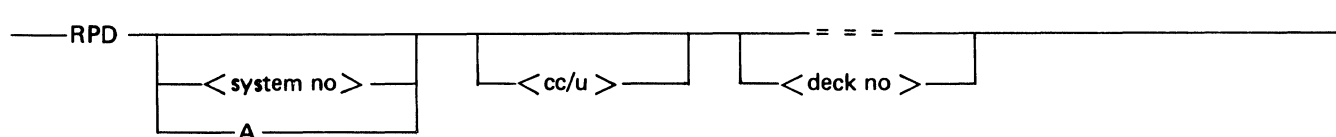


Figure 2-177B. RD, RDP Command Syntax (Format 2 - Pack)

Format 1

RD

indicates that the decks to be removed are on disk.

<system no>

selects pseudo decks created by a particular system in a multi-system shared configuration.

If <system no> is not supplied, the default is the system from which you enter the RD command.

A

The letter A selects pseudo decks created by all systems.

===

Three equal signs select all pseudo decks.

<deck no>

removes the pseudo deck with the specified pseudo deck number.

Pseudo decks are designated as #snnnn. The s represents the system number and is not required unless the deck was created by a system other than the one from which you are entering the command. The nnnn is a four-digit number with leading zeros assumed.

Do not include the pound sign in the command syntax.

Format 2

RDP

indicates that the decks to be removed are on pack.

<cc/u>

is a channel number/unit number and can be used to designate that only files from a specific pack are to be removed. You can specify **<cc/u>** only in an RDP command.

Details

The RD command removes only the pseudo decks that are not currently active.

You can display on the ODT the inactive pseudo card decks with the CD, CDP (List Inactive Pseudo Decks) command. An active deck can be inactivated by the DA, DAP (Deactivate Pseudo Reader) command, after which it can be removed by the RD command.

Examples

RD 5

removes pseudo deck 0005 from the system on which the command is entered.

RD 20035

removes pseudo deck 0035 from system 2.

RD 3 12

removes pseudo deck 0012 from system 3.

RD A ===

removes all inactive pseudo decks from all systems.

RDP 6/0 ===

removes all inactive pseudo decks from the pack that is channel 6/unit 0.

See Also:

CD, CDP (List Inactive Pseudo Decks); DA, DAP (Deactivate Pseudo Reader); and RN, RNP (Activate Pseudo Reader) commands.

REMOVE (Delete Files)

MCP Control Instruction

Access Level: 5

The REMOVE command deletes files from the disk or pack directory, making the file's space available.

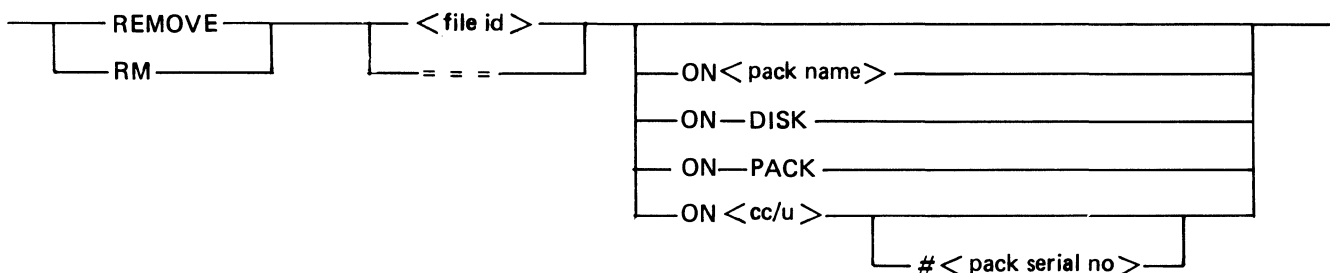


Figure 2-178. REMOVE Command Syntax

<file id>

===

is the name of the file to be removed. Equal signs (=) can be used as masking characters. Three or more equal signs (===) refers to all files.

ON <pack name>

is the name of the pack family on which the file is located. For example, ON MYPACK.

ON DISK

indicates that the file is on 100-byte-per-sector disk. This is the default.

ON PACK

indicates that the file is located on 180-byte-per-sector pack. All nonrestricted packs will be searched for the file.

ON <cc/u>

indicates that the base area of the file is on a pack drive with the specified channel number and unit number. For example, ON 07/3.

ON <cc/u> # <pack serial no>

removes the areas of the file that are on a pack drive with the specified channel number and unit number and whose base area is a pack with the specified serial number. Masking characters are not allowed in the file name.

CAUTION

Be careful with the REMOVE <file name> ON <cc/u> # <pack serial no> syntax, as it is possible to remove part of a file and leave a partial file on the pack. This syntax is provided for recovery from pack failures, not for normal use.

Restrictions

Files in use are not removed. Files for which library maintenance procedures are not allowed are not removed. This includes certain system files and certain files controlled by ALLOCATE records in the system configuration file.

A null mix is required to remove all files from disk or pack.

A comma (,) is required between file identifiers if the REMOVE command is entered from the ODT, or from the card reader with a SPO command.

Security

This restriction applies only if the SECURITY is present in the system configuration file.

If a file has a security type of PRIVATE or GUARDED, the file can be removed only by those allowed access to the file, or by users with LIBMAINT security privileges

System security is discussed in the *V Series System Software Security Installation and Operations Guide*.

Duplicate Files On Pack

If a masking character is used in the file name and files are found with duplicate names, all the duplicate files are removed, if possible.

If no masking characters are used in the file name and if two or more files with the same name are found on different families, no files are removed and the DUP FILE message is displayed on the ODT. You can then use the ON <pack name> syntax to remove the file you want.

If two or more files with the same name are found on the same pack family, you can use the ON <cc/u> syntax to remove the file you want.

File Areas On Pack

Pack files are divided into file areas. Each area occupies contiguous space on the pack. The areas of a file are not necessarily contiguous to one another. In a pack family with multiple packs, the areas of a file can be on different packs in the family. However, a file is never split between several families. The purpose of breaking a file into areas is to allow more efficient use of the packs.

The first area in a file is known as area one or the base area. If one pack in a multipack family has been designated as the master pack, the base area of all files on that family will go on the master pack.

If the ON <pack name> syntax is used, the base area of the file will be sought on the given pack family. If the base area is not found, no areas are removed, and the NO FILE message is displayed on the ODT.

If the ON <cc/u> syntax is used, only those files are removed whose base areas are on the given pack unit. If the base area is not found, no areas are removed, and the NO FILE message is displayed on the ODT.

If a file is being removed but some of its continuation areas cannot be found because of missing packs, the available continuation areas will be removed. The base area will not be removed, but will be flagged that a remove is in progress. No further access is allowed to the file. When the remaining continuation areas (and the base area) are present, do another REMOVE command. When all continuation areas are removed, the base area will be removed also.

The ON <cc/u> # <pack serial no> syntax is intended for the case in which the pack containing a file's base area is destroyed. It is the only way to remove continuation areas of a file without having access to the base area. Any areas on the given pack will be removed, whether they are the base area or continuation areas.

Details

This command works by removing the file's name and attributes from the disk or pack directory and header and returns the space occupied by the file's areas.

Examples

REMOVE FILEX

removes the file named FILEX from disk.

RM A123, OUPR, CDE ON DISK

removes the files named A123, OUPR, and CDE from disk.

REMOVE RS= ON PACK

removes any files whose names begin with the letters "RS" that are on any nonrestricted pack.

REMOVE R==S ON PACKA

removes any files whose names are five characters long, start with the letter R and end with the letter S, and are on the pack named PACKA.

REMOVE FILEA ON MYPACK, BOB11 ON DISK

removes the file named FILEA from the pack named MYPACK and also removes the file named BOB11 from disk.

REMOVE AFILE ON 06/2

removes the file named AFILE from the pack that is unit 2 on channel 6.

See also:

KX (Obtain User Disk Space); MR (Remove Duplicate File); and RM (Remove Duplicate File) commands.

The ALLOCATE record in *V Series MCP/VS System Software Operations Guide Volume 1: Installation*. System security is discussed in the *V Series System Software Security Installation and Operations Guide*.

RERUN (Log a Rerun Operation)

MCP Control Instruction

Access Level: 1

The RERUN command is used with a job initiation command, such as EXECUTE or COMPILE. It marks the job as a rerun operation in the system Run log (RLOG). The RERUN command has no effect on the scheduling or execution of any program.

RERUN

Figure 2-179. RERUN Command Syntax

Effect On The Run Log

This command sets the one-bit in the supplementary execution code field (RL-SXE, which is digit 71) of the job's Job Schedule record (type 3/0), the Short Job Schedule record (type 4/0), and Beginning-of-Job record (record type 6/0) in the RLOG.

Use

If one program starts another program using the ZIP facility, the RERUN command can be included in the ZIP syntax. In this way, an entire system of programs can be correctly flagged in the RLOG, as well as the one originally executed.

Examples

EXECUTE APROG; RERUN

starts the program called APROG and marks it as a rerun operation.

COMPILE MYPROG WITH COBOL9; RERUN

compiles the program called MYPROG and marks the compilation as a rerun operation.

See also:

TEST (Mark A Test Operation) command.

The Run log in the *V Series System Software Logging Operations Reference Manual*.

The ZIP facility is discussed in the various language manuals.

RF, RFP (Remove Backup File From Disk or Pack)

Keyboard Input Message

Access Level: 5

The RF command removes printer backup files, punch backup files, and memory dump files from disk or pack.

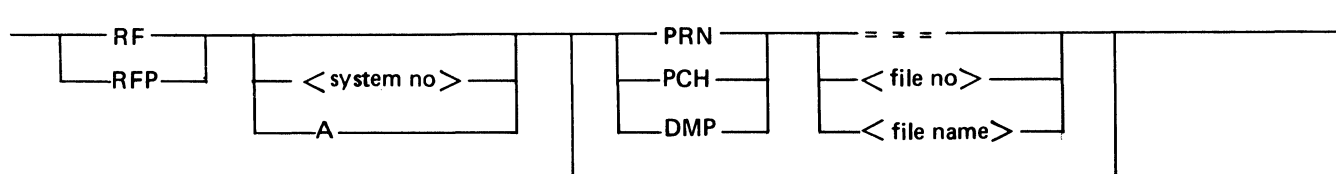


Figure 2-180. RF, RFP Command Syntax

RF
RFP

RF removes files from disk. RFP removes files from pack.

<system no>

A

<system no> indicates that only files created by that system are to be removed. The A means all systems. The default is the system through which the RF command is entered.

PRN
PCH
DMP

select the type of file to be removed. The choices are:

- PRN to remove printer backup files
- PCH to remove punch backup files
- DMP to remove memory dump files

Note that you cannot use RFP with dump files, as dump files are not written to pack. If you do, the message "****KBD IGNORED: INV SYNTAX" will be displayed.

===

removes all files. If PRN, PCH, or DMP is given, all files of that type are removed. If no file type is given, files of all three types are removed.

<file no>

is the backup file number or dump file number of the file to be removed. The \$, @, or * that prefixes the file number is not included. Since the first digit of the file number is the system number of the system that created the file, this number can be dropped if the file was created by the system from which the command is entered, or if the system number is otherwise given. Leading zeros can be dropped.

<file name>

is the backup file name (internal name) of a printer backup file, punch backup, or dump file. For PRN and PCH files, the name is a multifile-identifier or file identifier; for DMP files, the name is a program name that can be of the form multiprogram-id/program-id.

Restrictions

This command does not remove files that are in use.

Examples

RF PRN == =

removes all printer backup files from disk.

RFP PRN 32

removes the pack printer backup file whose backup file number is @s0032, where s denotes a system number. If the command is entered at system 2, for example, the file @20032 is removed.

RFP PRN 20032

removes the pack printer backup file whose backup file number is @20032. This is a file created on system 2, no matter which system the command is entered from.

RFP PRN RPRT

removes the pack printer backup file whose name is RPRT.

RF DMP 2

removes the dump file whose number is \$s0002, where s denotes a system number. If the command is entered at system 2, for example, the file \$20002 is removed.

RF PRN ABC=

removes all disk printer backup files whose names begin with the letters ABC.

NOTE

The intrinsic BFRF either lists or lists and removes backup files according to a wide range of options. See BF, BFP (List Backup Files) for more information.

See also:

BF, BFP (List Backup Files) and BFRF (Search, List, Remove Backup Files) commands.

RM (Remove Duplicate File)

Keyboard Input Message

Access Level: 5

The RM command removes a file from disk or pack to resolve a DUP LIB situation. The new file will remain intact.

A DUP LIB situation occurs when a program is trying to create a file but a file with that name already exists. The RM command removes the file that already exists.

`<mix no> RM`

Figure 2-181. RM Command Syntax

`<mix no>`

is the mix number of the task that is creating the file you want to save.

Details

The RM command removes the "old" file and leaves the "new" file, while the MR (Remove Duplicate File) command removes the "new" file and leaves the "old" file.

The RMOV option, if set, can resolve DUP LIB conditions on disk or pack by indicating that the old file is to be removed.

Example

The following is a scenario of messages displayed on the ODT in which the RM command is used. (These messages would display on your system only if the OPEN, CLOS, BOJ, and EOJ system options were set.)

ODT Display	Comment
==> EXECUTE JOB1	The operator starts a program called JOB1.
BOJ JOB1=012 012586 9:15	JOB1 starts to execute and is assigned a mix number of 012.
DSK FILE FILEA OPEN OUT JOB1=012	JOB1 creates a file called FILEA. It is a disk file and has been opened for output.
DSK FILE FILEA CLOSED RELEASED JOB1=12	JOB1 closes the file FILEA.
==> EXECUTE JOB2	The operator starts a program called JOB2.
BOJ JOB2=018 103185 9:20	JOB2 starts to execute and is assigned a mix number of 018.
DSK FILE FILEA OPEN OUT JOB2=018	JOB2 creates a file which happens to be called FILEA and which is a disk file.
**DUP LIB FILEA JOB2=018	JOB2 tries to close the file FILEA. The MCP detects that there is already a file called FILEA. JOB2 waits.
==> SPO 018 RM	The operator enters the RM command. This will remove the version ? FILEA that has been created by JOB1, allowing the "new" version of FILEA to close.
DSK FILE FILEA CLOSED RELEASED JOB2=018	JOB2 is now able to close its file.
EOJ JOB2=018 9:30 IN 1:36 CHG 2:22	JOB2 continues executing until it finishes.

See also:

MR (Remove Duplicate File) and REMOVE (Delete Files) commands.

The USE RMOV record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

RN, RNP (Activate Pseudo Reader)

Keyboard Input Message

Access Level: 1

The RN command processes the control records in a pseudo deck on disk or pack.

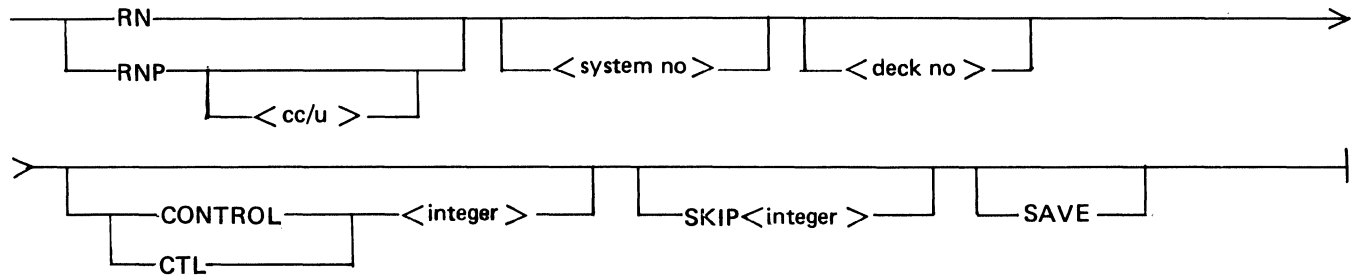


Figure 2-182. RN, RNP Command Syntax

RN
RNP

RN indicates that the pseudo decks to be activated are on disk; RNP, on pack.

<cc/u>

can be used to designate that only decks from a pack with the specified channel number and unit number are to be activated.

<system no>

selects pseudo decks created by a particular system in a multi-system shared configuration.

If <system no> is not supplied, the default is the system from which you enter the RN command.

=

indicates that all decks are to be activated. For example, RN = will activate all available readers on disk and on pack as well. RNP = will activate the available readers only on pack.

<deck no>

indicates that an individual deck is to be activated.

Pseudo decks are designated as #snnnn. The s represents the system number and is not required unless the deck was created by a system other than the one from which you are entering the command. The nnnn is a four-digit number with leading zeros assumed.

Do not include the pound sign in the command syntax.

**CTL
CONTROL**

are synonymous and allow positioning to any control statement within the pseudo deck. If you omit the CONTROL clause, processing begins with the first card in the reader.

<integer>

specifies the card number of the control card where point deck processing is to begin. This number can be obtained from the information supplied by the MCP when the PCRM option is set.

The PCRM option, if set, causes the display of the location and contents of the first control card and each DATA/DATAB in a control card group as they are processed from pseudo decks.

SKIP <integer>

designates the number of data records to be skipped before the first data file in the pseudo reader is to be assigned.

SAVE

specifies that the deck is to be retained when processing is completed; otherwise, the reader is purged when all statements have been read.

NOTE

A pseudo deck created by a CANDE "COPY TO CTLD" process can be activated only on the system on which it was created, and while the CANDE system is running.

Examples

RN =

activates all pseudo card decks on disk.

RN 20002

activates pseudo card deck number 20002, which is on disk.

RN 18 SAVE

activates pseudo card deck number 18 and saves the deck after processing is finished.

RNP 2

activates pseudo card deck number 2, which is on pack.

See also:

CD, CDP (List Inative Pseduo Decks); DA, DAP (Deactivate Pseudo Reader); OL (Display Peripheral Status); RD, RDP (Remove Pseudo Decks); and SD (Set Deck Limit) commands.

The LIMIT READERS, USE PCRM, and USE APCR records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

RO (RESET SYSTEM OPTIONS)

Keyboard Input Message

Access Level: 8

The RO command resets (turns off) system options.

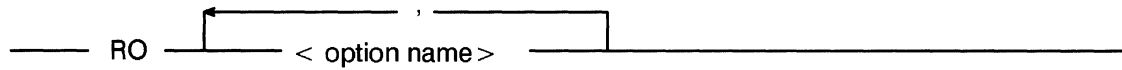


Figure 2-183. RO Command Syntax

<option name>

specifies a system option that was set either through one of the USE records of the system configuration file or through the SO (Set System Options) command.

See Table 2-12 under SO (Set System Options) for the available options to be reset.

Details

To see the status of a system option, you can use the TO (Type System Options) command.

Resetting the DEBUG, DCOM, DCP, DMS2, MICR, QWIK, SYST and TRAK options causes the MCP to release the memory space used by the designated extension module. These modules can be loaded at any time. However, certain modules, such as MICR, are not released until all users of the modules no longer require that module.

The Remote Job Entry (RJE) module is loaded at system initialization time because the MCP requires portions of it to handle Work Flow Language (WFL) programs. Resetting the RJE option causes the MCP to disable only the RJE handler portion of the RJE module.

Examples

RO DCOM

resets the DCOM option that loads a data comm module into memory when a data comm device is activated. The MCP will release the memory space used by the DCOM module.

RO TPLB, OPEN, CLOS

resets the TPLB option that displays a message each time a tape file becomes available; the OPEN option that displays a file open message each time a program opens a file; and the CLOS option that displays a file closed message each time a program closes a file.

See also:

SO (Set System Options) and TO (Display System Options Status) commands.

The USE records in the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

RP (READY AND PURGE MAGNETIC TAPE)

Keyboard Input Message

Access Level: 7

The RP command combines the RY (Ready Peripheral Device) and PG (Purge Magnetic Tape) commands to make a magnetic tape unit ready and to purge the tape.

Figure 2-184 shows the syntax for the RP command. However, the RP command also has additional capabilities that allow you to assign the tape to a mix number when purged and/or to override a program's tape type request. The syntax for these capabilities is shown in Figure 2-185.

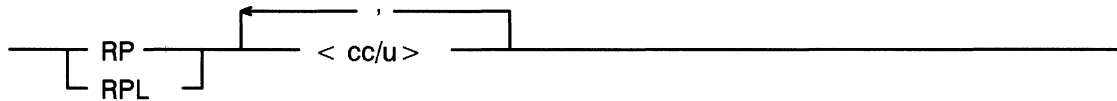


Figure 2-184. RP Command Syntax (Format 1)

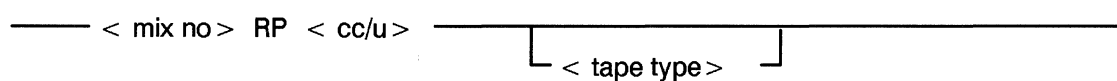


Figure 2-185. RP Command Syntax (Format 2)

Format 1

RP
RPL

indicates that you want to ready and purge the specified tape unit. Use RPL to lock the tape unit after it is purged, preventing its inadvertent assignment to a program.

<cc/u>

is the channel number/unit number of the magnetic tape unit to be readied and purged.

Format 2

<mix no>

represents the mix number of a job. If the specified job is waiting for an output tape, the tape is purged and assigned to this job. You can specify a <mix no> only when using Format 2 and cannot specify RPL.

<tape type>

overrides a user program's tape type request and is allowed only when using a <mix no> RP command (that is, Format 2). You can specify the <tape type> as MT9, MPE, GCR or MTC.

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

Examples

RP 4/1, 4/2, 4/5

readies and purges the magnetic tapes on channel number 4, unit numbers 1, 2, and 5.

RPL 4/3

readies, purges, and locks magnetic tape unit number 3 on channel number 4.

113 RP 6/1 MPE

readies and purges the magnetic tape unit on channel number 6, unit number 1 with a tape type of MPE and assigns it to the job with mix number 113.

See also:

AC (Purge And Place Tape Number In Label); PG (Purge Magnetic Tape); RY (Ready Peripheral Device); and SN (Purge And Place Tape Number In Label) commands.

RQ (Remove Storage Queue Entries)

Keyboard Input Message

Access Level: 8

RQ purges the storage queue (STOQ) of all queue entries in a specific queue or all entries in all queues.

The contents of the storage queue can be determined through the WQ (Display STOQUE Count) command.

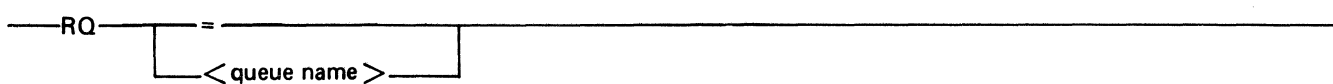


Figure 2-186. RQ Command Syntax

=
removes all entries from all queues.

<queue name>
is the name associated with a group of STOQ entries.

Explanation of Storage Queue (STOQ)

A storage queue (STOQ or STOQUE) is a buffer in memory used to transfer information asynchronously from one program to another. This means that if a program sends a message to another program through STOQ and that program is not ready to accept the message, the message is stored in the storage queue until the other program is ready to retrieve it.

The receiving program need not be in the mix when the message is sent. The receiving program retrieves its messages, one at a time.

A storage queue is created when a program puts a message in it or requests a message from it. It continues to exist as long as there are messages in it, or as long as a program is awaiting a message from it.

Each storage queue has a name, assigned by the programs according to the rules of the programming language. The sending program and the receiving program must, in general, agree on this name.

Examples

RQ =
removes all entries from the STOQ.

RQ JACK
removes the group of STOQ entries associated with the queue named JACK.

See Also:

WQ (Display STOQUE Count) command.

Complete details on storage queues are given in the various language manuals and in the *V Series MCP/VS Programming Reference Manual*.

RS (Remove Job From Schedule)

Keyboard Input Message

Access Level: 1

The RS command removes a scheduled job request from the MCP schedule list.

<mix no> RS

Figure 2-187. RS Command Syntax (Remove Specific Job)

<mix no>

designates a job that is in the schedule but not yet in the mix.

If the AFTR system option is set, all programs to be executed after the indicated program and all successors to these jobs are also removed.

If AFTR is not set, successors to the job being removed with an RS are not disturbed.

Removing All Scheduled Jobs

To remove all scheduled jobs from the MCP schedule list, use the syntax shown in Figure 2-188.

RS =

Figure 2-188. RS Command Syntax (Remove All Jobs)

Examples

12 RS

removes job with schedule number 012 from the MCP schedule list.

RS =

removes all scheduled jobs.

See also:

RA (Remove After-Linkage) and WS (Display Jobs in Schedule) commands.

The USE AFTR record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

RUN (Initiate Execution Of Program)

MCP Control Instruction

Access Level: 1

The RUN command initiates the execution of a program under timesharing.

—RUN<program name>—

Figure 2-189. RUN Command Syntax

<program name>

must be the name of a program, up to six characters long, in the system Process Information Block (PIB); that is, previously declared in a PROGRAM record in the CANDE initialization file.

Details

The program is subject to the standard restrictions applying to timesharing processes, except that it is not attached to any terminal.

Programs initiated with RUN are not placed in the schedule but enter the mix immediately. If timesharing is terminated, any programs running under timesharing are terminated.

Programs initiated through Work Flow Language (WFL), Remove Job Entry (RJE), and BNA network architecture cannot be run in the timesharing area. RUN is invalid with RJE and BNA and initiates a batch-area execution with WFL.

Example

RUN CPTEST

executes the program named CPTEST under timesharing.

See Also:

The *V Series CANDE Installation And Operations Guide* for information on timesharing.

RW (Rewind and Unload Tape)

Keyboard Input Message

Access Level: 7

The RW command unloads a tape residing on the specified tape drive. If the drive is declared SHARED, it will also be saved. The unload function works only for tape drives that have the automatic unload feature.

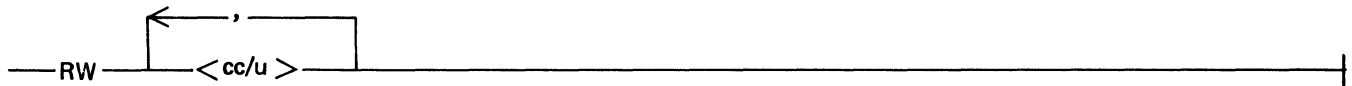


Figure 2-190. RW Command Syntax

<cc/u>

is the channel number/unit number of the magnetic tape unit.

Examples

RW 6/1, 6/2

saves and unloads the tapes on channel 6, unit numbers 1 and 2.

See Also:

SV (Save A Peripheral Unit).

Access Level: 8

You can use the **WX (Display Removed Space)** command to list all areas of memory, disk, or pack that have been removed from the system.

Returning memory:



<length>

are in thousands of digits (KD) and can specify all or part of an area removed by the XM command.

Returning disk space:



is the unit ID number of the disk.

<length>

must specify the beginning address of an area removed by an XD command or no space will be returned. If there is a match, the entire space of the removed area is returned independently of the optionally specified `<length>`.

The limits for <begin address> are 0-999999; the limits for <length> are 1-999999. Both <begin address> and <length> are always specified decimally.

Returning pack space:

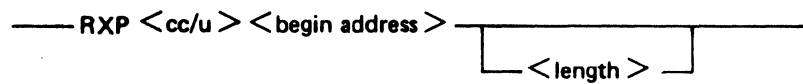


Figure 2-193. RX Command Syntax (Pack)

<cc/u>

is the channel number/unit number of the pack unit.

<begin address>

<length>

the <begin address> must specify the beginning address of an area removed by an XP command or no space will be returned. If there is a match, the entire space of the removed area is returned independent of the optionally specified <length>.

The limits for <begin address> are 1-16700000; the limits for <length> are 1-16700000. Both <begin address> and <length> are always specified in decimal.

Examples

RXM 155 4

returns 4KD at location 155000.

RXD 3 5275

returns the space on disk with ID 3 at address 5275.

RXP 1/9 2500

returns the space at address 2500 on pack on drive 1/9.

See also:

WX (Display Removed Space); XD (Remove Disk Segments); XM (Remove Memory From System); and XP (Remove Pack Sectors) commands.

RY (Ready Peripheral Device)

Keyboard Input Message

Access Level: 7

The RY command clears the status of an unassigned peripheral unit and makes it available to the MCP.

The RY command resets all exception flags maintained by the MCP for the specified unit and makes locked or saved units accessible. The MCP then readies the device as applicable. For example, if the unit is a tape, it reads the tape label. If the unit is a train printer, it loads the translate table file. If the unit is a card reader, it reads the first card.

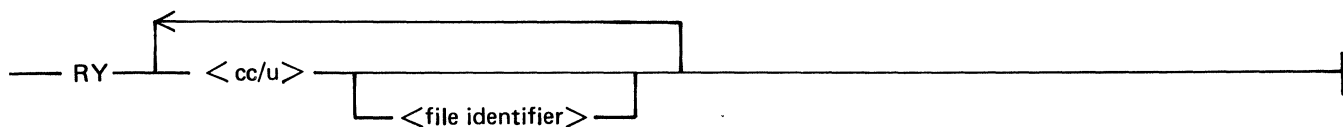


Figure 2-194. RY Command Syntax

<cc/u>

is the channel number/unit number of the peripheral unit to be made available.

<file identifier>

is for train printers and specifies the name of the translate table file to be loaded into the train printer buffer. File identifiers can be up to six characters long and cannot consist of all nulls, all blanks or all zeros.

You can have your own translate table files or use those bound to the MCP.

An RY command to a train printer always reloads the current (or specified) train buffer file.

An RY command to a train printer in use can specify only the train buffer file currently in use on the printer. This makes it impossible to accidentally change the train format during a print run.

NOTE

When a shared tape unit is made ready on two systems at the same time, any job using the tape will incur a rewind error and is discontinued.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

RY 10/2

clears the status of the unassigned peripheral on channel 10, unit number 2.

RY 3/4, 3/5, 1/0

clears the status of the unassigned peripherals on channel number 3, units numbers 4 and 5; and the peripheral on channel number 1, unit number 0.

RY 16/0 PRN123

clears the status of the unassigned train printer on channel number 16, unit number 0 and loads the translate table file called PRN123 into the train printer buffer.

See also:

RP (Ready and Purge Magnetic Tape); PG (Purge Magnetic Tape); and SV (Save a Peripheral Unit) commands.

SB (Interrogate Database Activity)

Keyboard Input Message

Access Level: 8

The SB command allows you to obtain information about database activity. You can display information about the following:

- one particular active database
- all jobs currently accessing a particular active database
- all active databases on the system

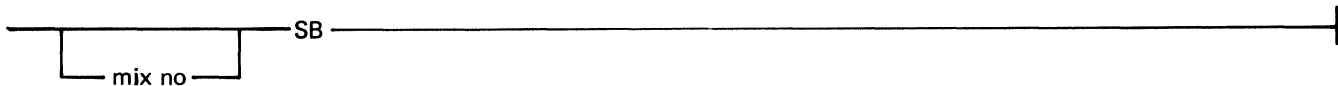


Figure 2-195. SB Command Syntax (Specifying Mix Number)

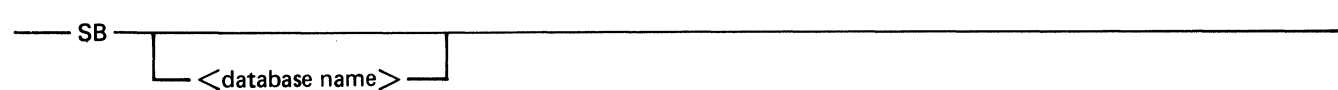


Figure 2-196. SB Command Syntax (Specifying Database Name)

SB

specified by itself without the <mix no> or <database name> options causes the system to display the names of all the currently active databases.

NOTE

Do not specify both <mix no> and <database name>; the presence of one excludes the use of the other.

<mix no>

causes the status of the active database program associated with that database user program to be displayed.

<database name>

causes the status of the active database program and all its active database user programs to be displayed.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Examples

Input: 12 SB
Output: P=7 M=7 DMS/DATAB1=032 WTG DMS FUNCTION

Input: SB TESTDB
Output: P=4 M=4 PROGD1=020 EXECUTING
P=4 M=4 TESTV1=005 WTG DMS FUNCTION

Input: SB
Output: SB= DATAB1 TESTDB DBACC5

See also:

SP (Display/Set/Change Database Parameters) command.

The V Series DMSII Operations Reference Manual Volume 2: Administration.

SD (Set Deck Limit)

Keyboard Input Message

Access Level: 8

SD limits the number of decks that the MCP Automatic Pseudo Card Reader (APCR) function can have active at one time.

SD <integer>

Figure 2-197. SD Command Syntax

<integer>

can be a value from 0 to 80 and indicates the maximum number of readers that APCR can have active simultaneously.

If an SD command is not supplied to the system, the default value of 80 is assumed.

Note that the specified value is not affected by a halt/load operation.

Details

The APCR option can be set through the USE APCR record of the system configuration file or with an SO (Set System Options) command. The APCR option directs the MCP to automatically place pseudo card decks into pseudo card readers as the decks become present on disk.

If the APCR option is not set, use the RN, RNP (Activate Pseudo Reader) command to manually accomplish this function.

You can use the DA, DAP (Deactivate Pseudo Reader) to deactivate pseudo card readers.

Example

SD 60

limits the number of decks the APCR function can have active at one time to 60.

See also:

DA, DAP (Deactivate Pseudo Reader); RN, RNP (Activate Pseudo Reader); SO (Set System Options) commands.

The LIMIT READERS and USE APCR record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

SHOW (DISPLAY SYSTEM INFORMATION)

Keyboard Input Message

Access Level: 3

The SHOW command can display information on the ODT about the following:

- the default family names for printer backup files, next printer and punch backup numbers to be assigned, current low printer and punch backup numbers, current records per block and records per area values for printer backup files.
- one entry in a Block Lockout Table (BLT), or an entire BLT
- all channels connected to an exchange
- individual disk or pack files
- all system tasks active on the system
- the name, percent CPU available, and amount of memory available for each system in multi-system shared configuration
- one or all Shared Systems Processors (SSPs) on the system
- individual jobs, or groups of jobs, running on the system
- internal data structures used by the V Series Communication System (VCS)

— SHOW < option> —

Figure 2-198. SHOW Command Syntax

<option>

indicates one of the options on the following pages. These options can be abbreviated to any non-ambiguous abbreviation:

BACKUP

BACKUP

Figure 2-199. SHOW BACKUP Command Syntax

- displays the current primary and secondary backup families where backup files are stored by default.
- displays the next backup numbers to be assigned to printer and punch backup files.
- displays the lowest backup numbers used by operating system intrinsics (PBDPRN, BFRF and so forth) when searching for printer and punch backup files.
- displays the number of records in each block (PRNRPB) and each area (PRNRPA) of printer backup files.

BLT

displays an entry from a Block Lockout Table (BLT), or displays an entire BLT. A Block Lockout Table is a reference table showing which disk/pack blocks are locked by which programs.

BLT

< BLT no >

Figure 2-200. SHOW BLT Command Syntax

<BLT no>

indicates a specific Block Lockout Table entry.

EXCHANGE

displays all the channels on an exchange and some information about them.

EXCHANGE < exchange no >

Figure 2-201. SHOW EXCHANGE Command Syntax

<exchange no>

indicates the number of the exchange (0-77) for which you want the system to display information.

FILE

displays information about a pack or disk file.

```
_____ FILE _____  
          | < family > / < file > |  
          | < file > ON < family > |
```

Figure 2-202. SHOW FILE Command Syntax

<file>

is the name of the pack or disk file.

<family>

is the family name of a pack file.

IRS

displays information about all active operating system tasks.

```
_____ IRS _____
```

Figure 2-203. SHOW IRS Command Syntax

PROCESSORS

displays information, including amount of memory available, percent CPU available and Host-name, for a non-shared system or for each system in a multi-system shared configuration.

```
_____ PROCESSORS _____
```

Figure 2-204. SHOW PROCESSORS Command Syntax

SSP

displays information about an SSP or all SSPs on the system.

```
_____ SSP _____  
          | < SSP channel no > |
```

Figure 2-205. SHOW SSP Command Syntax

<SSP channel no>

indicates a specific SSP.

TASK

displays information about a single task.

—— TASK < task no > —————|

Figure 2-206. SHOW TASK Command Syntax

<task no>

is the mix number (2-775) of the user task about which you want the system to display information.

TASKS

displays information about all user tasks.

—— TASKS —————|

Figure 2-207. SHOW TASKS Command Syntax

■ TRAK

displays the current TRAK specifications (determined by the most recently entered BT command). If no BT comand has been entered, a message is displayed indicating that data is not being inserted into the TRAK buffer.

—— TRAK —————|

Figure 2-207a. SHOW TRAK Command Syntax

VCS

displays information about internal data structures used by the V Series Communication System (VCS). This SHOW option is primarily used for maintenance of the VCS system.

Examples

SHOW IRS

displays information about all operating system tasks (independent runners).

SHOW TASKS

displays all user tasks.

See also:

ALTER (Change System Parameters); DBDR (Produce SSP Content Report); and DBSR (Produce Block Lockout Table Report) commands.

SI (Set Remote Device Identification)

Keyboard Input Message

Access Level: 8

The SI command changes the identification of a remote device.

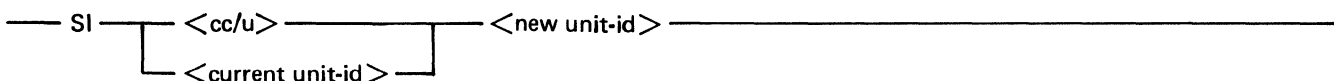


Figure 2-208. SI Command Syntax

<cc/u>

is the channel number/unit number of a data communications device.

<current unit-id>

is the current name of the device. The name can be one to six characters long. The first character must be alphabetic, but the remaining characters can be alphabetic or numeric.

<new unit-id>

is the name you want the device to have. The name can be one to six characters long. The first character must be alphabetic, but the remaining characters can be alphabetic or numeric.

Details

Note that a subsequent halt/load operation will reset the unit-id to its original UNIT record specification. See UNIT records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation* for more information.

Examples

SI 21/0 REMOT2

changes the remote device unit-id on channel 21, unit number 0 to REMOT2.

SI REMOT1 REMOT2

changes the remote device named REMOTE1 to REMOT2.

See also:

DL (Delete Peripheral Devices) and UNIT (Add Peripheral Devices) commands.

UNIT records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

SK (Skip Backup Records)

Keyboard Input Message

Access Level: 3

The SK command designates a number of records to be skipped during printing of a printer backup file (on tape or disk) or punching of a punch backup disk file.

You cannot use SK on a locked program.

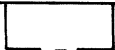
— <mix no> SK  <integer> —————

Figure 2-209. SK Command Syntax

<mix no>

identifies the PBDPRN or PCHOUT program that is to skip the number of records denoted by the <integer>.

—

is a minus sign, which when it precedes <integer> indicates that the records should be skipped backwards in the file. Up to -99999 records can be skipped backward.

<integer>

specifies the number of record to be skipped. Up to 999999 records can be skipped forward with each execution of this command.

NOTE

The number of records in the file is not necessarily equivalent to the number of lines of listing it produces.

Examples

007 SK -350

causes the PBDPRN or PCHOUT program associated with mix number 007 to skip back 350 records while printing or punching a backup file.

164 SK 1000

causes the PBDPRN or PCHOUT program associated with mix number 164 to skip forward 1000 records while printing or punching a backup file.

See also:

PC, PCP (Create Punch Backup File); PRINT (Invoke PBDPRN); and QT (Quit Program Operation) commands.

SM (SEND SYSTEM MESSAGE)

Keyboard Input Message

Access Level: 0

SM sends a message to a Message Control System (MCS) program or to controller programs of the V Series Communication System (VCS). The MCS programs can be for B 974, or B 874 data communications devices. They must be executing in the mix and have a queue attached. However, the queue cannot be full. The VCS controller programs include the Network Controller (NC) and the Application Controller (AC).

_____ < mix no > SM < text > _____

Figure 2-210. SM Command Syntax

<mix no>

is the mix number of the desired program. MCS programs retrieve text from their queues by issuing a READ MCS QUEUE function (function 81). VCS controller programs receive text through a "non-confirmed data indication" (type 22).

<text>

is the text of your message and can consist of any letters, numbers, and special characters except periods.

The maximum text length is 79 bytes.

Example

20 SM SYSTEM WILL BE COLDSTARTED AT 8 PM TONIGHT

sends the message to the MCS with mix number 020.

See Also:

SS (Send Message) command.

The *B 2000/B 3000/B 4000/V Series GEMCOS Implementation And Programming Reference Manual*.

SN (PURGE AND PLACE TAPE NUMBER IN LABEL)

Keyboard Input Message

Access Level: 7

The SN command purges a tape and puts a permanent identifying external physical tape number into a magnetic tape label.

The syntax for SN is shown in Figure 2-211. However, the SN command also has additional capabilities that allow you to assign the tape to a mix number when purged and/or to override a program's tape type request. The syntax for these capabilities is shown in Figure 2-212.

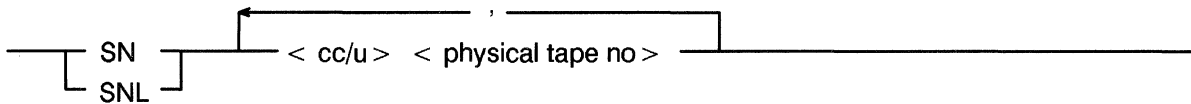


Figure 2-211. SN Command Syntax (Format 1)

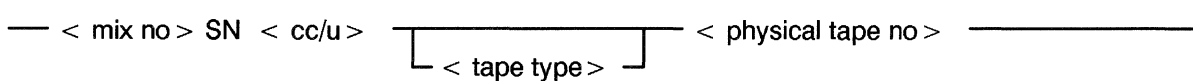


Figure 2-212. SN Command Syntax (Format 2)

Format 1

SNL

locks the tape after purging. SNL and <mix no> cannot be used in the same statement.

<cc/u>

is the channel number/unit number of the appropriate tape drive. The tape must be ready and contain a write ring.

<physical tape no>

is placed into the label of the designated tape reel and can be any integer up to five digits in length.

The <physical tape no> remains in the label throughout the life of the tape or until you perform another SN or AC operation on that tape.

Format 2

<mix no>

causes the tape on the drive designated by <cc/u> to be purged and assigned to the job with this mix number.

If the tape has GCR capability and no tape type is specified, GCR mode will be used.

<tape type>

overrides a user program's tape type request (only allowed when <mix no> SN is used). The <tape type> can be MT9, MPE, GCR or MTC.

Details

If you use format 1 (omitting the <mix no> and <tape type>), and you use a GCR-capable drive, scratch tapes are written in GCR mode.

Examples

SNL 4/1 22

purges the magnetic tape on channel number 4/unit number 1, places the number 22 in its label, and then locks the tape.

SN 46/1 123, 46/2 456

purges the magnetic tape on channel number 46/unit number 1 and assigns it the number 123; purges the magnetic tape on channel 46/unit number 2 and assigns it the number 456.

29 SN 12/3 GCR 84127

purges the GCR magnetic tape on channel 12/unit number 3, assigns it to mix number 029, and places the number 84127 in its label.

See also:

AC (Purge And Place Tape Number In Label); CN (Display Tape Number); LABEL1 (Change Tape Label); PG (Purge Magnetic Tape); RP (Ready And Purge Magnetic Tape); and TM (Write Tape Mark) commands.

SO (SET SYSTEM OPTIONS)

Keyboard Input Message

Access Level: 8

The SO command sets (turns on) certain system options.

System options can also be set through the USE records of the system configuration file. See for the V Series MCP/VS System Software Operations Guide Volume 1: Installation more information.

You can reset (turn off) system options with the RO (Reset System Options) command or see the status of a system option with the TO (Display System Options Status) command.

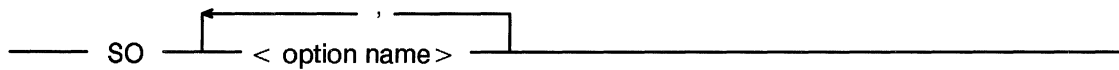


Figure 2-213. SO Command Syntax

<option name>

specifies one or more of the system options described in Table 2-12.

Table 2-12. System Options Available For Setting With SO

Option	Function
AFTR	automatically removes scheduled jobs that are after-linkaged to an abnormally terminated job
APBD	automatically prints and saves backup files after they are closed
APCR	automatically places pseudo card decks into pseudo card readers as the decks become present on disk or pack
AUHL	automatically initiates a halt/load in case of system failure
AURD	automatically removes pseudo card readers from disk if the files have erroneous control instructions
BOJ	displays Beginning-Of-Job (BOJ) messages each time a job enters the active mix
CHRG	specifies that all programs must have a charge number in order to execute
CLOS	displays a file-closed message each time a program closes a file
COPY	displays a message each time SYSTEM/COPY compares or transfers a file

Table 2-12. System Options Available For Setting With SO (continued)

Option	Function
DATE	requires a date be entered in order to halt/load
DEBUG	loads the interactive DEBUG module to trace the operating system or a user program
DCOM	loads a data comm module into memory when a data comm device is activated
DCP	loads a Data Communications Processor (DCP) into memory
DMS2	loads the DMS2 extension module into memory
EOJ	displays an End-Of-Job (EOJ) message each time a program goes to an EOJ
GTDK	forces program trace requests to backup disk
ISC	allows programmatic use of the Inter-System Control (ISC) for inter-system communications
LIB	displays a message each time a file is removed or changed
MICR	loads a reader-sorter module into memory
OPEN	displays a file open message each time a program opens a file
PBD	routes or directs printer backup files to disk
PBP	routes or directs printer backup files to pack
PBPS	directs all areas of printer and punch backup to a single pack
PBT	directs printer backup files to a scratch tape or a tape already in use for backup on a ready drive
PCD	indicates that punch card backup files are to be built on disk
PCH	directs punch files to the card punch without creating backup files
PCP	indicates that card backup files are to be routed to pack
PCRM	displays the location and contents of the first control card and each DATA(B) in a control card group as they are processed from pseudo card decks

Table 2-12. System Options Available For Setting With SO (continued)

Option	Function
PRN	directs printer files to the printer. No printer backup file is made
QWIK	loads the QWIK module which causes user program overlays to be stored and retrieved from a memory buffer
RJE	enables the Remote Job Entry (RJE) handler to be executed when a RJE station becomes active
RMOV	resolves duplicate file conditions on disk or pack by indicating that the old file is to be removed
RTSL	indicates that short or long tape block reads that are not multiples of the logical record size are to be treated as I/O errors
SCHM	causes a message to be displayed each time a program is scheduled for execution
SNAP	produces a SNAP pictures to capture the state of memory when certain processor errors occur
STGO	indicates that a message is to be displayed whenever an executing program stopped (such as with the ST or STOP (Suspend Program Processing) command), or when the program is restarted (such as with the GO or START (Resume Stopped Program) command).
SYST	loads a module into memory in order to run FLAME
SYUP	executes the SYSUP program after a cold start or halt/load
TERM	terminates programs that have irrecoverable errors, otherwise they will go into a DS/DP condition.
TIME	requires that the time be entered before a cold start or a halt/load completes
TPLB	causes a message to be displayed each time a tape file becomes available
TPMK	saves all tapes containing non-rewound reels with a write ring in the event of a halt/load
TRAK	loads a TRAK module into memory and allocates a buffer for tracking information
TRMD	dumps jobs that terminate abnormally
WRKP	directs work files to unrestricted system resource packs
ZIPM	displays the message "<job id> ZIP <control information message>" each time a program passes an instruction to the MCP

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Details

Setting the DBUG, DCOM, DCP, DMS2, MICR, QWIK, SYST and TRAK options causes the MCP to load an extension module into memory. These modules can be loaded at any time.

The RJE extension module is always loaded during system initialization because the MCP requires portions of it to handle Work Flow Language (WFL) programs. However, when you use SO to set the RJE option, the MCP enables the RJE-handling portions of the module and ensures that the DCOM module, which is also needed for RJE operations, is loaded as well.

Examples

SO PRN

sets the PRN option that causes print jobs to be printed so that no printer backup files need to be created.

SO TPLB, ZIPM, PBD

sets the TPLB option to have a message be displayed each time a tape file becomes available; the ZIPM option to have the "<job id> ZIP <control information message>" message be displayed each time a program passes an instruction to the MCP; and the PBD option to route or direct printer backup files to disk.

See also:

RO (Reset System Options) and TO (Display System Options Status) commands.

The USE records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

SP (Display/Set/Change Database Parameters)

Keyboard Input Message

Access Level: 8

Use the SP command to display, set, or change the value of certain database parameters while the Database Program (DBP) is executing.

The DBP is the program that performs all functions necessary for accessing and updating the database to which it is attached. All user application programs for a particular database interface with the DBP, and the DBP interleaves the processing of each program as much as possible.

For a complete description of the syntax and usage of this command see the *B 2000/B 3000/B 4000/V Series DMSII Operations Reference Manual Volume 2: Administration*.

See also:

SB (Interrogate Database Activity) command.

The *V Series DMSII Operations Reference Manual Volume 2: Administration*.

SPO (Simulate SPO Through Card Reader)

MCP Control Instruction

Access Level: 0

The SPO command provides capability for keyboard input messages and text to go to the MCP through a card reader as if they had been entered at the ODT. This resembles the ZIP construct allowed in an executing program by the various programming languages.

Any response to the message is sent to the ODT.

SPO <text>

Figure 2-214. SPO Command Syntax

<text>

can be any valid keyboard input message, except for the following restrictions:

- If a ? SPO card is entered in a card deck (including CTLDCK), it is processed immediately and does not cause EOF action on the executing program.
- ? SPO HL is invalid.
- SPO is invalid in a pseudo reader.

Details

The most common usage of SPO and its counterpart, the CC (Enter Control Instructions) command, is to send control information from a user program to the MCP using the ZIP facility or SPOMESSAGE (SPOM) BCT. ZIP assumes control text, so to enter a keyboard input message, you must prefix it with SPO. SPOM assumes keyboard input, so control text must be prefixed with CC. You can use the SPO and CC commands to switch from one mode to the other: ZIP "SPO RY 6/2; CC PFM LIST TAPE1 A."

Examples

? SPO LD

sends a LD (Create Pseudo Card Deck) command to the MCP. LD schedules the LDCNTL utility, which reads card files and converts them into pseudo card files on disk or pack.

? SPO RY 6/0

sends a RY (Ready Peripheral Device) command to the MCP to clear the status of the peripheral on channel number 6, unit number 0 and make it available to the MCP.

CALL SYSTEM ZIP USING "SPO SV 7/0, 16/0."

is a line of code from a COBOL74 user program that causes the operating system to execute the system command contained within the object program. In this case, the SPO command sends a SV (Save a Peripheral Unit) to the MCP. This SV command will save the peripheral devices associated with channel number 7, unit number 0 and channel number 16, unit number 0.

See also:

CC (Enter Control Instructions) command.

SQ (Squash Disk)

Keyboard Input Message

Access Level: 8

The SQ command instructs the MCP to consolidate all disk files into a contiguous area starting at the lowest address of the disk on which they currently reside. In this way, all available disk space on each disk is made contiguous.

This command is not available for MCP/VS 2.0.

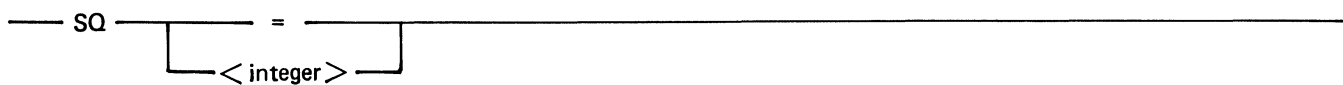


Figure 2-215. SQ Command Syntax

=

causes all disks to be squashed.

<integer>

instructs the MCP to squash files only the disk with the specified ID number. Disk devices must be assigned a disk ID number from 1 to 89.

Restrictions

To use this command, there must be a null mix.

Files declared through an ALLOCATE record in the system configuration file and certain system-created files are not moved.

Example

SQ 33

squashes the files on disk ID number 33.

See also:

SQP (Squash Pack) command.

SQP (Squash Pack)

Keyboard Input Message

Access Level: 8

The SQP command invokes the SQUASH utility, a bound intrinsic program that can perform SQUASH and/or COMPLEMENT functions on a designated pack as follows:

- SQUASH consolidates all areas on a designated pack.

When an area is moved, the table links, directory, and file headers are appropriately updated.

- COMPLEMENT reconstructs the Working Available Table from the information in the Master Available Table and the directory after the squash phase is complete. It can also be invoked as a single function. See the VALUE specifications for more information.

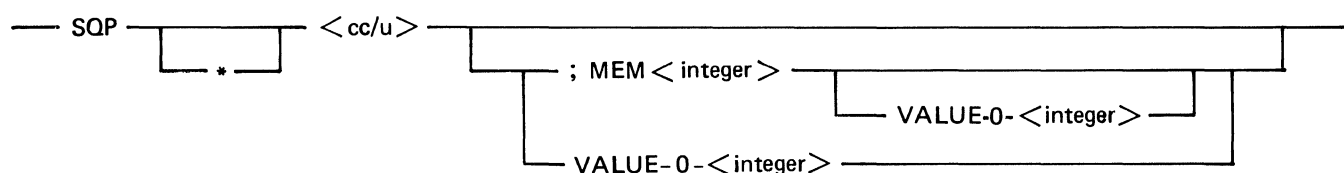


Figure 2-216. SQP Command Syntax

*

indicates that a recovery tape is to be created or loaded before the squash of the pack actually begins. This tape consists of the pack label, master available table, working available table, directory table, all file headers, and all file areas on the pack.

<cc/u>

designates the channel number/unit number location of the pack to be squashed or complemented.

MEM

requests additional memory to be allocated for execution of the SQUASH program. Adding memory to the minimum required by SQUASH speeds up the SQUASH function. It also slightly increases the speed of the tape dump and of the COMPLEMENT function. A value expressed by three or fewer characters is assumed to be KD (for example, 80 = 80KD); a value expressed by four to six characters is assumed to be digits (for example, 80000 = 80,000 digits).

VALUE 0

allows you to request a pack squash or pack complement by setting SW5 of the SQUASH program prior to its execution. The possible entries for VALUE are as follows:

- 0: Perform squash and compare any recovery tape created or loaded.
- 1: Skip comparison of any recovery tape created or loaded.
- 2: Create and compare recovery tape but do not squash pack.
- 3: Only create recovery tape. Do not compare tape or squash pack.
- C: Force pack complement even if the pack label indicates that pack recovery is required.
- D: Load pack from recovery tape ignoring the current recovery state of the pack but validating the recovery tape header record against the pack label. Thus, the pack label cannot be smashed for this to work.
- F: Only perform a pack complement.

Notes On The Use Of SQP

The pack must not be in use while a SQUASH operation is attempted. If any files are open on the designated pack when SQP is entered, the MCP will display the following message:

**** KBD IGNORED: UNIT IN USE**

When SQUASH is executed, it immediately saves the entire pack. The operator should not save the pack before entering SQP. No other programs can then access the pack until SQUASH completes.

In the case of processor or program exceptions, the program will automatically be reinstated at its own recovery routine to ensure a memory dump for recovery if you discontinue the SQUASH program.

SQUASH should not be attempted on a pack where a PS= indicates errors. Only DISPKV can clear up such a situation. See the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities* for more information.

SQUASH is not permitted on a shared pack. If SQP is entered for a shared pack, the MCP will respond with the message:

****INVALIDUNIT.**

Examples

SQP 18/2

invokes the SQUASH utility for a pack with a cc/u of 18/2.

SQP * 18/03; MEM 80000 VALUE 0 0

invokes the SQUASH utility for a pack with a cc/u of 18/3, allocates another 80000 digits of memory, and requests that a recovery tape be created, the pack squashed, and the tape compared.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

See also:

SQ (Squash Disk) command.

The DISPKV utility in *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

SS (Send Message)

Keyboard Input Message

Access Level: 0

The SS command directs messages from one ODT to another.

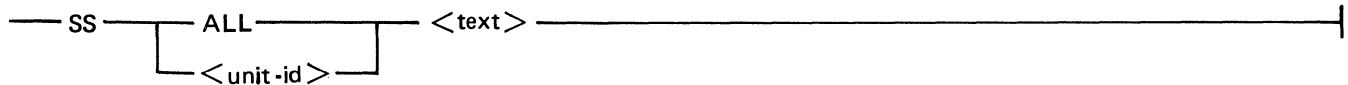


Figure 2-217. SS Command Syntax

ALL

directs a message from an ODT to all other ODTs on the system.

<unit-id>

specifies the name of the destination ODT. The name should be one declared in the UNIT ODT or UNIT OCS record of the system configuration file and should be one to six characters long. The first character must be alphabetic, but the remaining characters can be alphabetic or numeric.

<text>

is the text of your message.

Example

SS ALL GO OFF LINE PLEASE

sends the message "GO OFF LINE PLEASE" to all ODTs on that line.

See also:

SM (Display System Message) command.

The UNIT ODT and UNIT OCS records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

ST (Suspend Program Processing)

Keyboard Input Message

Access Level: 3

The ST command stops the execution of a program and makes the memory it occupies the prime candidate for rollout to disk if another program is waiting memory.

Rollout to disk is not done automatically. The program is removed from the ready to run list, and will be rolled out only if memory is required by a task.

The ST command has the same effect as the STOP (Suspend Program Processing) command.

The stopped program can be restarted with a GO (Restart a Stopped Program) or START (Restart a Stopped Program) command.

`<mix no> ST`

Figure 2-218. ST Command Syntax

`<mix no>`

is the mix number of the program to be suspended.

Details

Values in use by the program are saved, for use when it is restarted.

The MCP suspends execution of the designated program when all I/O operations in progress for that program are completed.

The execution of a GO (Restart A Stopped Program) command or START (Restart A Stopped Program) causes the MCP to resume execution of the program.

If the program to be stopped is not currently executing, the ST command is ignored.

Example

32 ST

stops the program associated with mix number 032.

See Also:

GO (Restart A Stopped Program); START (Resume Stopped Program); and STOP (Suspend Program Processing) commands.

START (Restart A Stopped Program)

MCP Control Instruction

Access Level: 3

The START command restarts a program that was stopped by the ST (Suspend Program Processing) or STOP (Suspend Program Processing) command.

The GO (Restart A Stopped Program) command has the same effect as the START command.

—START <program name>—

Figure 2-219. START Command Syntax

<program name>
is the name of the program to be restarted.

Details

The program restarts its execution at the point where it stopped. It starts with the same values it was using when it stopped.

If you issue START on a program that is not stopped, the system ignores the command.

If There Is Not Enough Memory

If there is not enough memory for the program being restarted, these rules apply:

- Memory is allocated by priority. If you raise the program's memory priority, it might restart.
- When sufficient memory becomes available, the program will restart.
- A program being restarted will start before a program of the same priority that is waiting in the schedule.
- Programs with lower memory priority are rolled out from memory to disk if this will make enough room for the program that is waiting to restart.

Programs with Duplicate Names

If two or more programs with the same name have been stopped, the START command starts the one with the highest processor priority. If the stopped programs have the same priority, it is unpredictable which will be restarted.

Example

START PAYROL
restarts a program named PAYROL.

See Also:

GO (Restart A Stopped Program); ST (Suspend Program Processing); and STOP (Suspend Program Processing) commands.

STOP (Suspend Program Processing)

MCP Control Instruction

Access Level: 3

The STOP command stops an executing program and makes the memory it occupies a prime candidate for rollout to disk if another program is waiting memory.

Rollout to disk is not done automatically. The program is removed from the ready to run list, and will be rolled out only if memory is required by a task.

The STOP command has the same effect as the ST (Suspend Program Processing) command.

— STOP <program name> —

Figure 2-220. STOP Command Syntax

<program name>

is the name of the program to be stopped.

Details

Values in use by the program are saved, for use when it is restarted.

This command takes effect when all I/O operations for the program have completed. Peripheral devices in use by the program are not released to the system.

The execution of a GO (Restart A Stopped Program) command or START (Restart A Stopped Program) causes the MCP to resume execution of the program.

If the program to be stopped is not currently executing, the STOP command is ignored.

If two or more programs with the same name are active when you issue the STOP command, the one with the highest priority will stop. If priorities are equal, it is unpredictable which one will stop.

Example

STOP PAYROL

stops a program named PAYROL.

See Also:

GO (Restart A Stopped Program); START (Resume Stopped Program); and STOP (Suspend Program Processing) commands.

SV (Save a Peripheral Unit)

Keyboard Input Message

Access Level: 7

The SV command saves a peripheral unit, preventing the system from accessing it.

The unit can be made ready again by a halt/load or with an RY (Ready Peripheral Device) or UR (Inhibit or Uninhibit Unit) command.

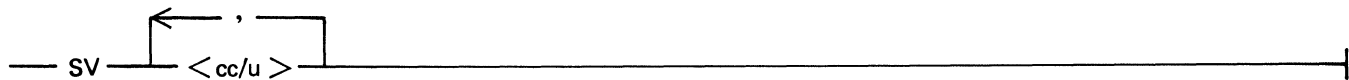


Figure 2-221. SV Command Syntax

<cc/u>

indicates the channel number and unit number of the peripheral to be saved.

Details

If a program is using a unit when the SV command is entered, the unit is not saved until the program releases it.

You can save a unit to prevent other programs from accessing it. Ready or uninhibit the unit when you want it to be used.

The SV command may be used to transfer a unit from one system to another, if the unit is physically connected to both systems. Save the unit on one system, and then ready it on the other.

Examples

SV 1/5

saves the unit on channel number 1/unit number 5.

SV 1/8, 1/5, 4/0

saves the unit channel 1/unit 8, channel 1/unit 5, and channel 4/unit 0.

See Also:

RY (Ready Peripheral Device) and UR (Inhibit Or Uninhibit Unit) commands.

SW (Set Programmatic Switches)

Keyboard Input Message

Access Level: 1

The SW command sets the programmatic switches in an executing program. These switches are features of some programs written in languages, such as COBOL.

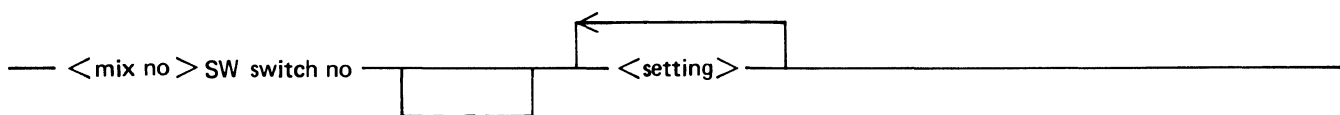


Figure 2-222. SW Command Syntax

<mix no>
is the mix number of the executing program.

<switch no>
is the number of the switch whose value you want to change. The switch number can be from 1 to 8.

=
is optional and does not influence the command.

<setting>
indicates the value of the switch setting and can be a number from 0 to 9.

More than one switch can be set by a single SW command. If <setting> is longer than one digit, the first digit sets the switch designated by the <switch no>. The subsequent digits sets the subsequent contiguous switches. For example, 012 SW 1 34 sets the switch 1 to a value of 3 and also sets switch 2 to a value of 4.

Restrictions

You cannot have a mix number of 0 or 1, because that indicates the MCP.

Switches cannot be set on a program that is locked. See the LOCK (Lock Function) command.

Details

Switches 0 through 8 correspond to COBOL switches SW0 through SW8.

This command inserts integer values into the executing program code. The values for SW8, SW1 through SW7 are inserted into program base-relative locations as shown in Table 2-13.

Table 2-13. Memory Locations of Programmatic Switches

Switch	Memory Location
8	000000
1	000001
2	000002
3	000003
4	000004
5	000005
6	000006
7	000007

The SW command cannot affect locations beyond location 7. If you need to do this, use the IN (Insert Data Into Program), INSERT (Insert Data Before Execution), or VALUE (Insert Value Into A Program) commands. IN can insert values past location 7 while a job is executing; INSERT and VALUE insert values prior to a job's execution.

Examples

2 SW 4 1

affects the job with mix number 002, setting switch 4 to a value of 1.

14 SW 2 109

affects the job with mix number 14, setting switch 2 to a value of 1, switch 3 to 0, and switch 4 to 9.

See Also:

IN (Insert Data Into Program); LOCK (Lock Function); VALUE (Insert Value Into A Program); and the discussions of <parameter 1> under the COMPILE (Compile Program), DEBUG (Attach Program To DEBUG Session), and EXECUTE (Execute Library Program) commands.

Programmatic switches are described further in the language manuals.

TEST (Mark A Test Operation)

MCP Control Instruction

Access Level: 1

The TEST command is used with a job initiation command such as EXECUTE or COMPILE. It marks the job as a test operation in the system Run log (RLOG). The TEST command has no effect on the scheduling or execution of any program.

TEST

Figure 2-223. TEST Command Syntax

Effect On The Run Log

This command sets the four-bit in the supplementary execution code field (RL-SXE, which is digit 71) of the job's Job Schedule record (type 3/0), Short Job Schedule record (type 4/0) and Beginning-of-Job record (type 6/0) in the RLOG.

Use

If one program starts another program using the ZIP construct, the TEST command can be included in the ZIP syntax. In this way an entire system of programs can be correctly flagged in the Run log, as well as the one originally executed.

Examples

EXECUTE PROG; TEST

starts the program called PROG, marking the job in the RLOG as a test operation.

See also:

RERUN (Log a Rerun Operation) command.

The system Run log in the *V Series System Software Logging Operations Reference Manual*.

The ZIP construct is discussed in the various language manuals.

The TI command displays on the ODT the amount of system time used by a particular job.

— <mix no> — TI —————

Figure 2-224. TI Command Syntax

Sample Response

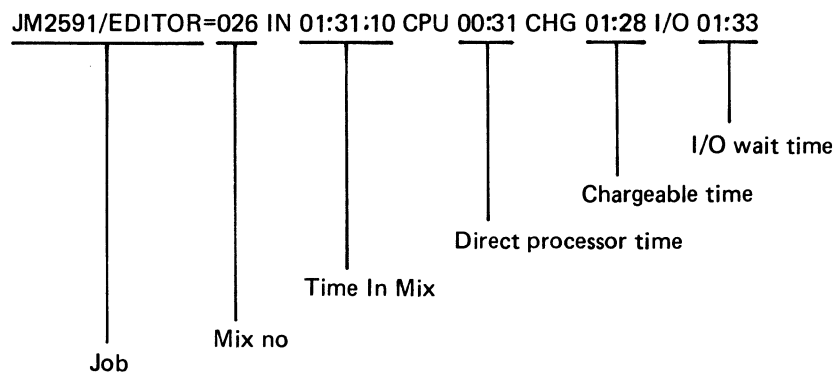


Figure 2-225. TI Display

TIME (DISCONTINUE PROGRAM AFTER A SPECIFIED AMOUNT OF TIME)

MCP Control Instruction

Access Level: 1

TIME discontinues a program (to EOJ) if it uses more than a given amount of processor time.

TIME = < integer >

Figure 2-226. TIME Command Syntax

=

is optional and does not influence the command.

<integer>

is the amount of processor time, in seconds, that the job is allowed to accumulate. It can be from one to five digits. If fewer than five digits are used, leading zeros are assumed.

Details

Processor time is the accumulated time the processor has spent working on the job. It is also called “direct time”.

You can use the TI (Type Time Statistics) command to find out the amount of system time used by a particular job.

Example

EXECUTE PAYROL; TIME 300

starts the program called PAYROL. If the job uses more than 300 seconds of processor time, the job will be discontinued. Note that there is no limit to the amount of time the job can be in the mix; only the amount of processor time is monitored.

See also:

TI (Display Time Statistics) command.

TL (TRANSFER LOG)

Keyboard Input Message

Access Level: 8

The TL command causes a log transfer. TL closes the current system log file and directs logging information into a new log file.

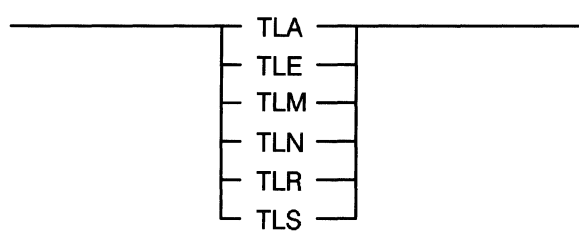


Figure 2-227. TL Command Syntax

■ TLA

transfers all logs in use - the Run Log, the Maintenance Log, the ODT Log, the Network Log and (for V 500 series systems) the ECM Log.

■ TLE

transfers the ECM Log. The Environmental Control Module (ECM) Log is created only on V 500 series systems.

TLM

transfers the Maintenance Log.

■ TLN

transfers the Network Log. The Network Log is used by the V Series Communication System (VCS).

TLR

transfers the Run log.

TLS

transfers the ODT log.

Use

Log transfer happens automatically when a log file becomes full or at LOGXFERTIME. The TL or LN (Transfer and Print Log) command can be used to close the current log file so you can analyze it, or when you want to start a new log file at a particular time. This depends on the information you are trying to capture.

The LN (Transfer and Print Log) command transfers the log and starts the log analysis program. LN always starts the log analysis program; TL starts it only if the AUTO option is set in the log's USE record in the system configuration file.

Details

The system configuration file can contain a USE record for each system log (USE RLOG, USE MLOG, USE NLOG, and USE SLOG). If a log's USE record is present, the log will be created.

If the AUTO option is set on a log's USE record, the appropriate log analysis and reporting program is started whenever that log is transferred, whether it is transferred automatically when it fills up or at LOGXFERTIME, or manually by the TL or LN command.

If the WRAP option is set on a log's USE record, log files are renamed each time they transfer by incrementing a number in the file name from 0 to 9. For example, the name of the MLOG would change from MLOGs1 to MLOGs9 (where s is the system number). If this option is not set, the file names do not change, and one log file will overwrite the other file at log transfer time. If this option is not used, the files will be renamed MLOGs0 to MLOGs1.

Examples

TLR

transfers the Run log.

TLA

transfers all logs in use.

See Also:

LN (Transfer and Print Log) command.

The USE LOG and CONTROL LOGXFERTIME records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *V Series System Software Logging Operations Reference Manual* for complete details on system logging.

TM (WRITE TAPE MARK)

Keyboard Input Message

Access Level: 7

The TM command writes a tape mark and, optionally, an End-Of-File (EOF) or End-Of-Reel (EOR) label at the current point on a magnetic tape.

—— TM < cc/u> —————

Figure 2-228. TM Command Syntax (Write Tape Mark Only)

The TM syntax used to write both a tape mark and an ending label is shown in Figure 2-229.

—— TM < cc/u> ————

FILE	< file id >			
REEL	< multifile id > / < file id >	< reel no >	< label type >	

Figure 2-229. TM Command Syntax

<cc/u>

is the channel number/unit number of the tape drive on which the tape is mounted. If this is all that is given, a tape mark is written.

FILE

writes a tape mark and an EOF label. This is the default if the file identifier is given.

REEL

writes a tape mark and an EOR label.

<file id>

<multifile id> /<file id>

these elements specify the name of the tape file. The name can be up to six characters (for example, MYFILE) or it can be a multifile id (for example, MYTAPE/MYFILE).

If either a file identifier or multifile id/file id is given, a tape mark and an EOF label are written.

<reel no>

is the number of the reel. The default is 1.

<label type>

specifies the kind of end label to write and allows the following values:

B: Unisys BSL standard, which is the default.

U: USASI standard

I: a label standard for your installation (see the INSTALLATION LABEL1 record in *V Series MCP/VS System Software Operations Guide Volume 1: Installation*)

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

Restriction

The tape drive must be in a saved state for this command to be used. Tape drives are automatically saved during halt/load if the TPMK option is set.

Uses

The TM command can be used to add tape marks or labels to a tape that lacks such delimiters because of a system failure during file creation.

After a halt/load, if the TPMK option is set, all tape drives with reels that need rewinding and have a write ring will have a saved status. You can use the TM command to write tape marks and labels, allowing these saved tape files to be properly closed.

Examples

TM 3/2

writes a tape mark at the current point on the tape on channel 3/unit number 2.

TM 10/3 REEL FILEA 003

writes a tape mark, an EOF mark for FILEA, and an EOR label for reel number 003 on channel 10/unit number 3. The label will be the default type.

See also:

AC (Purge And Place Tape Number In Label); CN (Display Tape Number); LABEL1 (Change Tape Label); SN (Purge And Place Tape Number In Label); and SV (Save a Peripheral Unit) commands.

The INSTALLATION LABEL1 and USE TPMK record in the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

TO (Display System Options Status)

Keyboard Input Message

Access Level: 0

The TO command returns the status of the system options.

System options can be set either through the USE records in the system configuration file or through the SO (Set System Options) command. The RO (Reset System Options) can be used to reset system options.

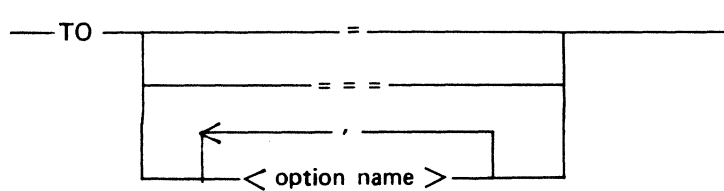


Figure 2-230. TO Command Syntax

=
displays the settings of system options that can be set and reset.

===
displays the settings of all system options.

TO <option name>
displays the setting of a particular option.

Examples

TO =
displays the setting of all settable/resettable system options, such as AFTR=1 APBD=1 COPY=1
RJE=0 SYUP=0 TRAK=1

TO OPEN
displays the setting of the OPEN option. For example: OPEN=0.

TO MICR, ZIPM, SNAP
displays the settings of the MICR, ZIPM, and SNAP options. For example: MICR=0 ZIPM=0
SNAP=1

See Also:

RO (Reset System Options) and SO (Set System Options) commands.

The USE records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

TR (Time Reset)

Keyboard Input Message

Access Level: 8

The TR command changes the time of day used by the MCP.

— TR <four-digit integer> —

Figure 2-231. TR Command Syntax

<four-digit integer>

represents what you want the new time to be. The time is given in 24-hour format (military time), without punctuation.

Restriction

No jobs can be in the mix when this command is given. If there are jobs in the mix, the command is ignored.

Examples

TR 0800
changes the time to 8:00 am.

TR 1535
changes the time to 3:35 pm.

See Also:

DR (Date Reset); WD (Display Date); and WT (Display Time) commands.

The USE DATE and TIME records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

TRACE (Initiate Trace)

MCP Control Instruction

Access Level: 3

The TRACE command indicates that the program is to be traced when it enters the mix.

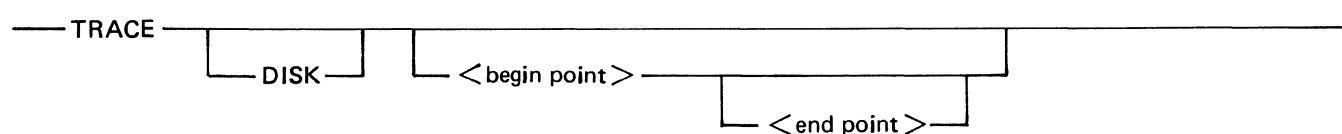


Figure 2-232. TRACE Command Syntax

DISK

specifies that the trace is to go to backup disk.

DISK has the same effect as using the OU (Direct File To Device) command to send the file to disk or as setting the GTDK system option. The GTDK option can be set through the USE GTDK record in the system configuration file or with an SO (Set System Options) command.

If this option is omitted, the trace goes to the system default medium, determined by the setting of the GTDK system option.

<begin point>

<end point>

allow the operator to specify beginning and ending limits for the trace printing.

If the <begin point> is specified, the trace begins printing when the program reaches the designated point.

If the <end point> is also specified, the printing terminates at that point and does not resume until the <begin point> is again reached. The program is still being traced during this period, but the printing is suppressed.

If you omit <begin point> and <end point>, tracing begins with the first instruction and ends with the last instruction.

The format for the <begin point> and <end point> specifications is shown in Figure 2-233.

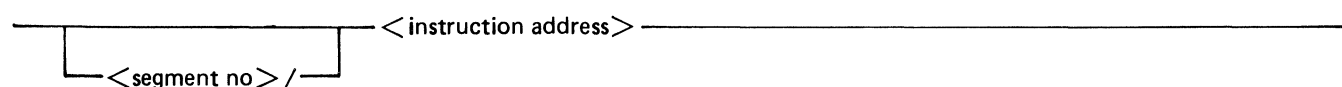


Figure 2-233. Begin/End-Point Specification Syntax

<segment no> /

<instruction address>

must be the exact segment number and address of an instruction. If either is 0, the corresponding limit is not enabled. A <segment no> of 0 means any segment, not just segment 0.

Details

The trace is listed on the first available line printer; if none is available, any printer in use by the designated program is used. If printers are not available, the program is suspended until a printer becomes available or until the operator directs the trace to backup disk through an OU (Direct File to Device) command. If the GTDK option is set, it goes to a disk backup file.

You can use the GT (Initiate Trace) command to trace a program that is executing.

The NT (Terminate Trace) command stops the tracing of a program.

Traces can also be started programmatically. The TRACE program branch communicate (BCT) requests the MCP to initiate or terminate a program trace. See the *V Series MCP/VS Program Interfaces Programming Reference Manual* and the various programming language reference manuals for more information.

The interactive DEBUG module has a number of sophisticated debugging features that can be used to trace user programs or the operating system itself. DEBUG (Attach Program To A DEBUG Session) and ID (Invoke DEBUG) can be used to start DEBUG sessions. GT and TRACE can be used only for user programs.

Example

EXECUTE PROG; TRACE

starts the program PROG and begins tracing the program starting with the first instruction.

See also:

DEBUG (Attach Program To A DEBUG Session); GT (Initiate Trace); ID (Invoke DEBUG); NT (Terminate Trace); OU (Direct File To Device); and SO (Set System Options) commands.

The USE GTDK record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *V Series Program Interfaces Programming Reference Manual* and the various programming language manuals for information about TRACE.

UL (Assign Unlabeled File)

Keyboard Input Message

Access Level: 7

The UL command tells the MCP where to find an unlabeled file that is needed by a program.

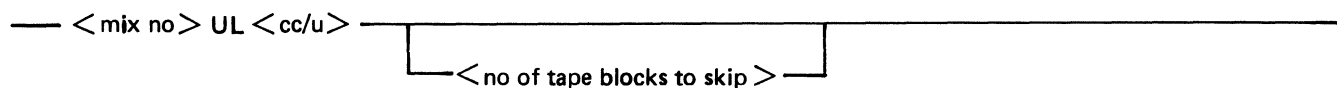


Figure 2-234. UL Command Syntax

<mix no>

is the mix number of the program waiting for a file.

<cc/u>

indicates the channel number/unit number of the device where the file is located.

<no of tape blocks to skip>

applies to magnetic tapes. When the program opens the file, the tape is spaced forward this number of blocks, or until a tape mark is encountered, whichever comes first. Then the first data block is read. This allows magnetic tapes prepared on non-Unisys systems to be read.

The number of blocks to skip must be less than 100.

Uses

This command can be used when a program cannot find a file that it needs, which causes a NO FILE message to appear on the ODT. You can direct the program to an unlabeled file by using the UL command. You specify a peripheral unit. The program expects the file on that peripheral unit to be the file it needs. It expects the file to be unlabeled. If the file is a tape with a foreign label not recognized by the system, the label may be bypassed with the skip count.

This command can be used to read magnetic tapes that lack standard tape labels. This includes tapes prepared on non-Unisys systems.

Hardware Types

The UL command can be used with:

- Magnetic tape files
- Card files
- Reader/sorter files

For card files, the recording mode is assumed to be EBCDIC.

Card readers and reader/sorters can be either ready or not ready when the command is given.

V Series MCP/VS System Software Operations Guide, Volume 2

The System Commands

Examples

23 UL 1/2

tells the program with mix number 023 that the unlabeled file it needs is on unit 2 of channel 1.

5 UL 0/5 2

tells the program with mix number 005 that the unlabeled tape it needs is on unit 5 of channel 0, and that the tape must be spaced forward two blocks (or to the first tape mark) before it is read.

See also:

FR (Final Reel); IL (Assign Labeled File); and OF (Indicate Optional File) commands.

UNIT (Add Peripheral Devices)

MCP Control Instruction

Access Level: 8

The UNIT command adds a peripheral device to the system.

Use the UNIT command for devices other than disks or packs, such as:

- 80-column card readers and card punch
- data communications processors (DCPs)
- buffered, image page printer, and train printers
- Inter-System Connects
- tape drives
- Non-Status devices (NSTs)
- Operator Display Terminals (ODTs) and Operator Control Stations (OCSs)
- data communications devices that connect to the system through a UNILINE DLP
- check reader sorters

Disks and packs can be declared at system initialization time; or added with the DISK (Add A Disk) or PACK (Add A Pack) command.

The syntax for the UNIT command resembles the syntax for the UNIT records of the system configuration file.

Adding An 80-Column Card Reader Or Card Punch

To add an 80-column card reader or 80-column card punch to the system environment, use the syntax shown in Figure 2-235.



Figure 2-235. UNIT Command Syntax (CRD and PCH)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

=

is optional and does not influence the command.

CRD

denotes an 80-column card reader.

PCH

denotes an 80-column card punch.

RETRY <0-99>

indicates the number of times the system should attempt to complete an I/O. The default setting is 10 retries. A value of 0 prevents any error recovery attempts.

SAVED

saves the drive when it is added and after cold start. The device remains saved until made available with a RY (Ready Peripheral Device) command.

RESERVED

reserves the unit when it is added and after cold start. This can be used to declare a non-functioning device to the system. You can use the UR (Inhibit or Uninhibit Unit) command to make the reserved device available.

Example

UNIT 3/0 = PCH RETRY 15 SAVED RESERVED

adds to the system an 80-column card punch, allows it 15 retry attempts, and gives the punch a saved and reserved status when it is added and after a cold start.

Adding A DCP

To add a Data Communications Processor (DCP) to the system environment, use the syntax shown in Figure 2-236.

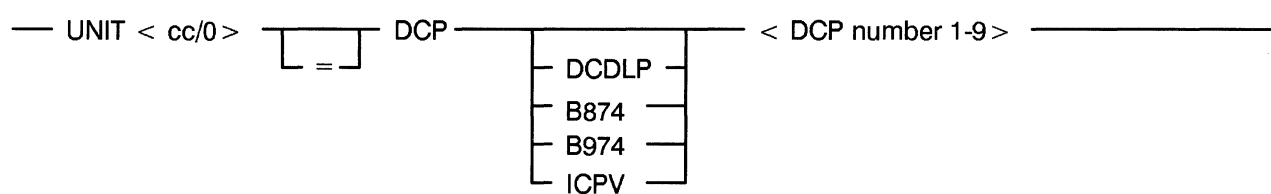


Figure 2-236. UNIT Command Syntax (DCP)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

=

is optional and does not influence the command.

DCP

denotes a Data Communications Processor.

DCDLP

B874

B974

■ ICPV

are used for documentation purposes only, since the exact model of DCP is determined automatically.

<DCP no 1-9>

indicates the logical DCP that is being declared. The number can range from 0 to 9, and this number is used to match each DCP specified in the NDL to a physical channel.

Example

UNIT 02/0 DCP B974 3

adds a B974 DCP with a DCP number of 3 to the system on channel 2.

Adding an NCP

To add a Network Communications Port (NCP) to the system environment, use the syntax shown in Figure 2-236a. NCPs are used as part of the V Series Communication System.

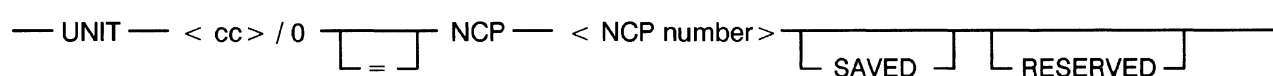


Figure 2-236a. UNIT Command Syntax (NCP)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

=

is optional and does not influence the command.

NCP

denotes a Network Communications Port.

NCP number

a 4-digit number assigned to the NCP in the VCS database.

SAVED

saves the NCP when it is added and after cold start. The NCP remains saved until made available with a RY (Ready Peripheral Device) command.

RESERVED

reserves the NCP when it is added and after cold start. This can be used to declare a non-functioning NCP to the system. You can use the UR (Inhibit or Uninhibit Unit) command to make the reserved NCP available.

Details

Under previous operating systems, the CP 3680 is declared with two UNIT declarations. When a CP 3680 communications processor is connected as part of an NCP, it needs only one UNIT declaration.

Example

UNIT 16/0 NCP 5 SAVED

adds an NCP to the system environment. The NCP is on channel 16, has an NCP number of 5 in the VCS database and is in saved status when it is added and after a coldstart.

V Series MCP/VS System Software Operations Guide Volume 2
The System Commands

Adding An Image Page Printer (IPP)

To add an IPP to the system environment, use the syntax shown in Figure 2-237.

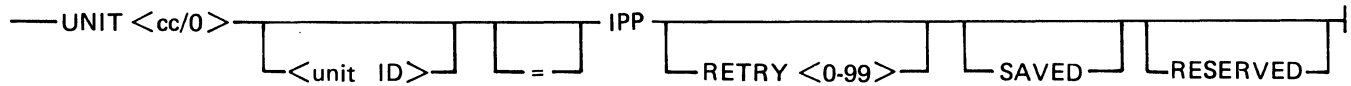


Figure 2-237. UNIT Command Syntax (IPP)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

<unit ID>

denotes the identifier of the unit. The IPP can have up to a six-character name. It is recommended that the unit have the same name as its default printer control file (PCF).

=

is optional and does not influence the command.

IPP

denotes an Image Page Printer.

RETRY <0-99>

indicates the number of times the system should attempt to complete an I/O. The default setting is 10 retries. A value of 0 prevents any error recovery attempts.

SAVED

saves the drive when it is added and after cold start. The device remains saved until made available with a RY (Ready Peripheral Device) command.

RESERVED

reserves the unit when it is added and after cold start. This can be used to declare a non-functioning device to the system. You can use the UR (Inhibit or Uninhibit Unit) command to make the reserved device available.

Example

UNIT 6/0 DOCPCF = IPP SAVED

adds to the system an IPP called DOCPCF on channel 6. When the IPP is added and after a cold start, the device will be saved.

Adding An Inter-System Connect (ISC)

To add an ISC to the system environment, use the syntax shown in Figure 2-238.

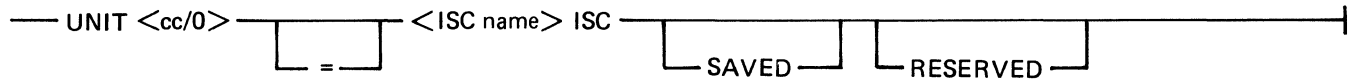


Figure 2-238. UNIT Command Syntax (ISC)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

=

is optional and does not influence the command.

ISC

denotes an Inter-System Connect (ISC).

<ISC name>

is a one- to six-character name. In BNA, it will be the first six characters of the IO-STATION-GROUP name and, therefore, the external name of the ISC file through which the station group is communicating.

SAVED

saves the drive when it is added and after cold start. The device remains saved until made available with a RY (Ready Peripheral Device) command.

RESERVED

reserves the unit when it is added and after cold start. This can be used to declare a non-functioning device to the system. You can use the UR (Inhibit or Uninhibit Unit) command to make the reserved device available.

Example

UNIT 1/0 ISC GEORGE

adds an ISC called GEORGE to the system on channel 1.

Adding MPE, GCR, MTC and MT9 Tape Drives

To add MPE, GCR, and MT9 type tape drives to the system, use the syntax shown in Figure 2-239.

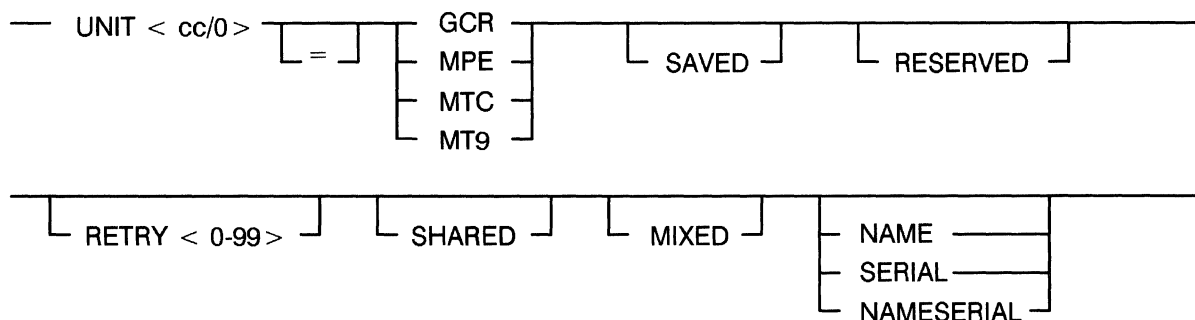


Figure 2-239. UNIT Command Syntax (MPE, GCR, MTC, MT9)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

=

is optional and does not influence the command.

GCR

denotes a tape drive that supports Group Coded Recording (GCR 6250 bpi) type tape.

NOTE

Drives that support more than one recording density must be declared as GCR. To select a mode other than GCR, use syntax within a user program or override the drive's default mode using system commands. For example, to write a PE mode tape on a GCR drive that can operate in PE mode, use the tape type override feature of the AC (Place Tape Number in Label), PG (Purge Magnetic Tape), RP (Ready and Purge Magnetic Tape) or SN (Place Tape Number in Label) commands.

MPE

denotes a tape drive that reads and writes Magnetic Phase-Encoded (MPE 1600 bpi) type tape.

MT9

denotes a tape drive that reads and writes MT9 (800 bpi) type tape.

MTC

denotes a tape drive that reads and writes MTC (38000 bpi) type tape cartridges.

SAVED

saves the drive when it is added and after cold start. The device remains saved until made available with a RY (Ready Peripheral Device) command.

RESERVED

reserves the unit when it is added and after cold start. This can be used to declare a non-functioning device to the system. You can use the UR (Inhibit or Uninhibit Unit) command to make the reserved device available.

RETRY <0-99>

indicates the number of times the system should attempt to complete an I/O. The default setting is 10 retries. A value of 0 prevents any error recovery attempts.

SHARED

saves the drive each time a program closes a tape file with a LOCK or RELEASE.

This automatically saves a drive after a halt/load.

SHARED is used when a drive is physically connected to more than one system in a multi-system shared configuration. The drive should never be readied by more than one system at any given time as access errors will occur.

MIXED

indicates that the device supports the MT9 and MPE tape densities. MIXED does not apply to GCR or MTC types.

NAME

SERIAL

NAMESERIAL

These options apply to MTC tape drives only. NAME indicates that you want the unit's display panel to display the tape name. SERIAL indicates that you want the unit's display panel to display the serial number of the tape. NAMESERIAL indicates that you want the unit's display panel to alternate the display between the tape name and the serial number.

Example

UNIT 04/0 MPE RESERVED RETRY 20

adds to the system an MPE tape drive on channel 4. When the device is added and after a cold start, the device will be reserved. Twenty retry attempts to complete an I/O are allowed.

Adding An NST

To add an NST to the system environment, use the syntax shown in Figure 2-240. NSTs should be used for diagnostic purposes only.

UNIT < cc/u > NST

Figure 2-240. UNIT Command Syntax (NST)

<cc/u>

indicates the channel and unit number of the device.

NST

specifies a Non-Status device.

Example

UNIT 6/3 NST

adds an NST to the system on channel 6, unit number 3.

V Series MCP/VS System Software Operations Guide, Volume 2 The System Commands

Adding An ODT or OCS

To add an Operator Display Terminal (ODT) or Operator Control Station (OCS) to the system environment, use the syntax shown in Figure 2-241.

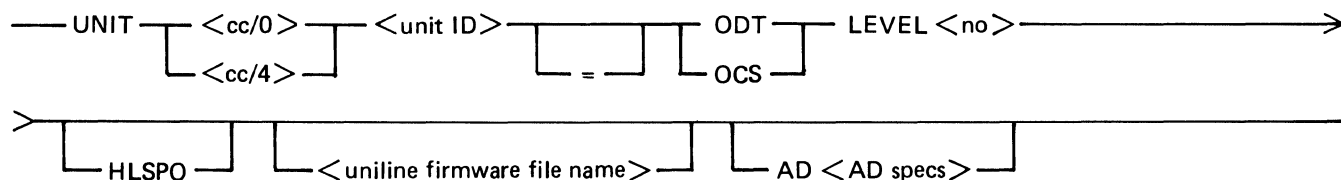


Figure 2-241. UNIT Command Syntax (ODT, OCS)

<cc/0>

<cc/4>

indicates the terminal's channel number and unit number. The unit number must be 0 for an OCS or 4 for an ODT.

<unit ID>

is a one- to six-character name. The first character must be alphabetic,

LEVEL <no>

limits the level of MCP commands that can be input from the terminal. See Access Level Restrictions, section 1, for more information.

HLSP0

indicates that the terminal is the HLSP0 and will display halt/load and system failure messages.

<uniline firmware file name>

indicates the name of the file that contains the UNILINE firmware, if the terminal is connected to a UNILINE DLP.

AD <AD specifications>

See the AD (Specify ODT Screen Parameters) command for information about the AD specifications that control the automatic display of system status information on an ODT.

Example

UNIT 4/4 A107 = ODT LEVEL 7

adds an ODT to the system. The ODT must have unit number 4 and will be on channel 4, have a unit ID of A107, and have an access level of 7. The default AD specifications will be used.

Adding A Buffered Or Train Printer

To add a buffered or train printer to the system environment, use the syntax shown in Figure 2-242.



Figure 2-242. UNIT Command Syntax (PRN, TPR)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

<unit ID>

denotes the train image file to be loaded.

For buffered printers, use the name PRN256 for the 2000 LPM printer, 1050-1250 printer, and 600 LPM printer.

For train printers, use the name of the default train printer image file (for example, (FUL96). The files can be copied and renamed to provide for unique names for the printers. See the **LOADER** utility in *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities* for more information.

=

is optional and does not influence the command.

PRN

denotes a buffered printer

TRP

denotes a train printer.

SAVED

saves the drive when it is added and after cold start. The device remains saved until made available with a **RY** (Ready Peripheral Device) command.

RESERVED

reserves the unit when it is added and after cold start. This can be used to declare a non-functioning device to the system. You can use the **UR** (Inhibit or Disinhibit Unit) command to make the reserved device available.

Example

UNIT 5/0 PRN256 PRN SAVED

adds a buffered printer to the system environment. The 2000 LPM printer will be on channel 5, has an ID of PRN256, and will have a saved status when it is added and after a cold start.

Adding Data Communications Devices (RJE, TC5, VDD, TWX)

To add a RJE, TC5, VDD, or TWX data communications devices to the system environment, use the syntax shown in Figure 2-243.

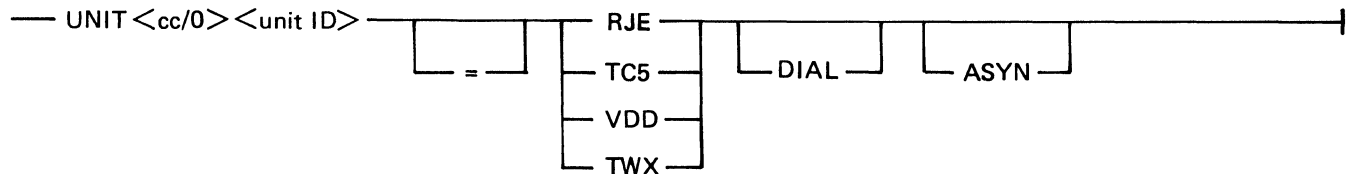


Figure 2-243. UNIT Command Syntax (RJE, TC5, VDD, TWX)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

<unit ID>

is a one- to six-character name for RJE, TC5, VDD, and TWX devices. The first character must be alphabetic, the remaining characters alphanumeric. Unlike most other devices, data communications lines have no label or other physical identifier that associates a line with the device. To allow programmatic assignment by name, each line is assigned a unit ID. The unit ID doubles as its adapter ID.

=

is optional and does not influence the command.

RJE

denotes a Remote Job Entry device.

TC5

denotes a TC 500 device.

VDD

denotes a B 20 or ET series workstation or a B 9352 or TD 800 series terminal.

TWX

denotes a teletypewriter through a UNILINE DLP.

DIAL

specifies that the devices is linked to a switched line.

ASYN

is used only with RJE devices and declares the line as asynchronous.

Example

UNIT 2/0 JGM33 RJE ASYN

adds to the system an RJE device, with a unit ID of JGM33, on channel number 2. The device will be using asynchronous communication.

Adding Check Reader Sorters

To add S4A or S4B check reader sorter devices to the system environment, use the syntax shown in Figure 2-244.



Figure 2-244. UNIT Command Syntax (S4A, S4B)

<cc/0>

represents the channel number and a unit number for the device. Note that the unit number of the device must be 0.

<unit ID>

denotes the identifier of the unit. The unit ID for S4A and S4B devices is a one- to six-character name. The first character must be alphabetic, the remaining characters alphanumeric.

=

is optional and does not influence the command.

S4A

denotes a 9137 reader sorter.

S4B

denotes a 9138 reader sorter.

SAVED

saves the drive when it is added and after cold start. The device remains saved until made available with a RY (Ready Peripheral Device) command.

RESERVED

reserves the unit when it is added and after cold start. This can be used to declare a non-functioning device to the system. You can use the UR (Inhibit or Disinhibit Unit) command to make the reserved device available.

Example

UNIT 4/0 S4A002 S4A RETRY 12 SAVED

adds to the system a S4A check reader sorter, with a unit ID of S4A002, on channel number 4. This unit is allowed 12 retry attempts and will have a saved status when it is added and after cold start.

See Also:

DISK (Add A Disk); DL (Delete Peripheral Devices); PACK (Add A Pack); RY (Ready Peripheral Device); and UR (Inhibit or Disinhibit Unit) commands.

The DISK, PACK, and UNIT records in the system configuration file in *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

UNLOAD (Move Library Files Between Media, Remove Old Files)

MCP Control Instruction

Access Level: 5

The UNLOAD command moves files. When the files have been moved to the new location, they are deleted from the old location.

We recommend that you use the MOVE command and SYSTEM/COPY instead of the UNLOAD command. The UNLOAD command will become unavailable at a future release.

The UNLOAD command starts a program called LOADMP, if disk files are to be moved to tape, or a program called PACKUP, if files are to be moved to or from pack. LOADMP and PACKUP must be written by the user; Unisys does not supply them. If the appropriate program is not present on disk, the UNLOAD command is ignored.

See Appendix B for UNLOAD syntax, if you have user-coded LOADMP and/or PACKUP utilities.

See Also:

COPY (File Transfer Utility) command. Appendix B.

UP (Unlock Program)

Keyboard Input Message

Access Level: 1

The UP command unlocks a locked program, thus allowing certain ODT commands to be directed to that program.

— <mix no> UP —————→

Figure 2-245. UP Command Syntax

<mix no>

is the mix number of the locked program to be unlocked.

Uses

This command reverses the effect of the LOCK or LP (Lock Program) commands. After the program is unlocked, it can be discontinued or otherwise manipulated.

Example

23 UP

unlocks the locked program associated with mix number 023.

See Also:

LOCK (Lock Function) and LP (Lock Program) commands.

UR (Inhibit or Uninhibit Unit)

Keyboard Input Message

Access Level: 8

The UR command can either temporarily remove a peripheral device from the system or restore it to the system or interrogate UR status.

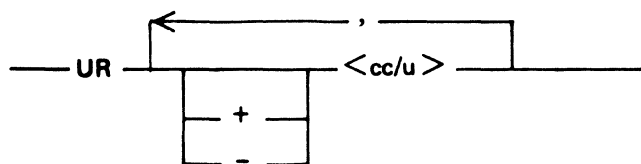


Figure 2-246. UR Command Syntax

UR (without + or -)
displays whether the device is currently inhibited or not.

UR +
inhibits the unit.

UR -
uninhibits the unit.

<cc/u>
is the channel number/unit number of the device.

Details

Inhibited status, caused by a UR + command or the RESERVED option on DISK, PACK, and UNIT records or commands, can be removed only by a UR - command

When a device is inhibited, it is ignored by the MCP and cannot be assigned to a program. However, it can still be reached by direct I/O.

Inhibited status is retained across halt/loads.

You should use the DL (Delete Peripheral Devices) command to permanently remove a device from the system.

A device that has been permanently removed from the system is reintroduced by the UNIT (Add Peripheral Devices) command.

Examples

UR + 4/3
inhibits the device on channel 4/unit 3.

UR - 3/0
uninhibits (makes available) the device on channel 3/unit 0.

UR + 2/0, - 23/1
inhibits the device on channel 2/unit 0 and uninhibits the device on channel 23/unit 1.

See Also:

DL (Delete Peripheral Devices) and UNIT (Add Peripheral Devices) commands.

USER (Assign Access Code)

MCP Control Instruction

Access Level: 0

The USER command assigns an access code to a single job or input command. An access code is a usercode/password/charge number combination.

You enter the USER command before a command that starts a program, such as COMPILE (Compile Program) or EXECUTE (Execute Library Program). The program then executes under the attributes of that access code. It can also be used to precede file maintenance commands or any other command.

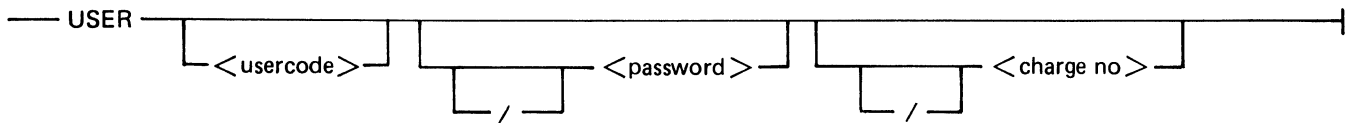


Figure 2-247. USER Command Syntax

<usercode>

refers to a code that identifies a user to the system.

<password>

refers a group of characters by which the system determines the legality of a user.

<charge no>

refers to a number used to assign charges for system use.

NOTE

The <usercode>, <password>, and <charge number> are all optional. Slashes are required only if leading parameters are omitted (for example, USER //63023).

Details

The device from which the USER command is entered does not have to be logged in.

The security attributes of the access code become attached to the single next program or other function started from this device. These attributes override any that might have been active from previous BEGINUSER (Beginning of Job Stream) or LI (Log In) commands. After the USER command has its effect, the security attributes revert to what they were before the USER command.

If the CHRG option is set, a charge number must be given for each program initiation (certain MCP utilities excepted).

The charge number can be overridden by the CHARGE command. CHARGE associates a program with a charge number but does not affect the user's or the program's security attributes. (However, the charge number along with the usercode and password must form a valid access code.)

Note that the BEGINUSER (Beginning of Job Stream) and LI (Log In) commands affect security attributes for all programs executed from the logged-in device, whereas USER affects them for only one program.

Example

USER USR2/PASS2/999999; EXECUTE RUST

starts the program RUST with the security attributes of usercode USR2, password PASS2, and charge number 999999.

See Also:

BEGINUSER (Beginning Of Job Stream); CHARGE (Insert Charge Number); ENDUSER (End Of Job Stream); LI (Log In); and LO (Log Out) commands.

The SECURITY and USE CHRGE records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

Details on system security are given in the *V Series System Software Security Installation and Operations Guide*.

VALUE (Insert Value Into A Program)

MCP Control Instruction

Access Level: 1

The VALUE command inserts six digits into the code file of a program prior to its execution.

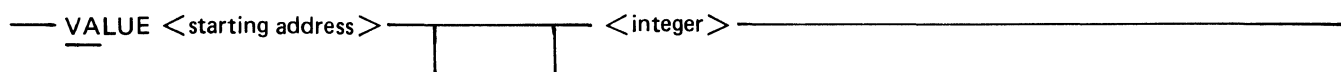


Figure 2-248. VALUE Command Syntax

<starting address>

is the program base-relative address at which to start inserting. This can range from 00000 through 00058.

Addresses 0-7 (memory locations 000000-000007) can represent programmatic switches as shown in Table 2-14.

Table 2-14. Memory Locations of Programmatic Switches

Switch	Memory Location
8	000000
1	000001
2	000002
3	000003
4	000004
5	000005
6	000006
7	000007

=

is optional and does not influence the command.

<integer>

can contain the digits 0 through 9 and the undigits A through F. Six digits are always inserted into the program. If <integer> is less than six digits, leading zeros are assumed.

Details

You can use the IN (Insert Data into Program) and INSERT (Insert Data Into Program) commands to enter information into a program. INSERT is used at program initiation time, while IN is used on a program that is already executing.

The SW (Set Programmatic Switches) command sets programmatic switches, which is equivalent to inserting values at locations 0 through 8. SW is used on a program that is already executing.

NOTE

The VALUE command cannot be associated with a COMPILE (Compile Program) or EXECUTE (Execute Library Program) commands if you use the <parameter list> option. This option replaces former MCP slash parameters. MCP/VS 2.0 continues to allow slash parameters. However, when they are used, a warning will be displayed on the ODT and written to the ODT log stating that slash parameters will be deimplemented at a future MCP/VS release.

The first parameter of the <parameter list> causes eight digits to be inserted, while VALUE inserts only six digits. This may necessitate the use of trailing zeros when VALUES that begin at address 0 or 1 are replaced by <parameter list>. For example: EX ABC VALUE 0 = 1234 is equivalent to EX ABC (123400); while EX ABC VA 2 = 5678 is equivalent to EX ABC (5678).

Example

COMPILE PROGA COBOL9; VALUE 0 = 100000

compiles PROGA with the COBOL9 compiler, inserts a value of 1 at location 0 of the COBOL9 compiler code, and inserts zeros at locations 2, 3, 4, 5, and 6.

See Also:

COMPILE (Compile Program); EXECUTE (Execute Library Program); IN (Insert Data into Program); INSERT (Insert Data Into Program); and SW (Set Programmatic Switches) commands.

WB (Display Bound Programs)

Keyboard Input Message

Access Level: 0

The WB command returns a list of the utility programs whose code is bound to (intrinsic to) the MCP.

—— WB = ——|

Figure 2-249. WB Command Syntax

Sample response:

The following two lines show a sample response:

```
WB = DMPOUT RLGOUT SLGOUT MLGOUT LDCNTL DSKOUT DMSDLR PRN256  
WB = COPY SQUASH DISPKV PBDPRN FUL96 FUL72 SORT: MERG: NEWLAB
```


WC (Display Available Memory)

Keyboard Input Message

Access Level: 0

The WC command returns the amount of contiguous memory, in thousands of digits (KD), that is currently available.

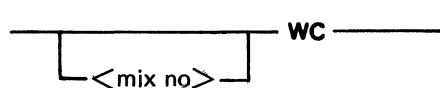


Figure 2-250. WC Command Syntax

<mix no>

If a mix number is supplied, the response is the amount of memory being used by that program, exclusive of disk file headers and device-alternate blocks.

Sample Response

```
MEMORY: 28463 77 50 30 30 30 15 9 9 8
         5 5 5 5 4 2 2 2 1 1
         1
```

TOTAL AVAILABLE: 28754 KD

In this response, there are 21 areas of memory available. One is 28463 kilodigits long and the others are 77, 50, 30, 30, 30, 15, 9, 9, 8, 5, 5, 5, 5, 4, 2, 2, 2, 1, 1, 1, kilodigits long. Each area is a piece of contiguous memory, but the areas are not contiguous to each other.

Details

Sometimes the MCP is not able to arrange memory so that all available memory is contiguous. This can be caused by memory that has been removed from the system by an XM (Remove Memory from System) command, among other causes.

If the response to WC indicates two or more areas of available memory, these are not necessarily the maximum amounts of memory that could be made available in each area. The MCP could increase the available space in one area and decrease the space in the other by moving programs from one area to the other. Thus, the decision to attempt the execution of a job must not be based solely on the response to a WC command.

As a historical note, WC stands for "what core".

Examples

WC

causes a display of the amount of memory currently available.

5 WC

causes a display of the amount of memory being used by the job with mix number 005.

See also:

XM (Remove Memory From System) command.

WD (Display Date)

Keyboard Input Message

Access Level: 0

The WD command returns either the current date or the day of the week of a given date.

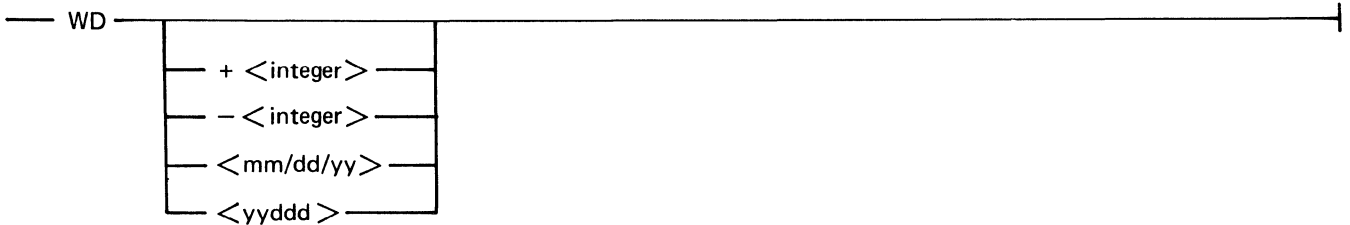


Figure 2-251. WD Command Syntax

WD

displays the current date, day of the week, and julian date.

+ <integer>

causes the number of days specified by <integer> to be added to the current date. The day of the week and the julian date are calculated.

- <integer>

causes the <integer> number of days to be subtracted from the current date. The day of the week and the julian date are calculated.

<mm/dd/yy>

causes a date to be supplied in month/date/year format - for example, 01/23/87 for January 23, 1987. The day of the week and the julian date are calculated.

WD <yyddd>

causes a julian date to be supplied, with two digits for the year and three digits for the day. For example, 85185 is the 185th day of 1985. The date and day of the week are calculated.

Sample Response

Figure 2-252 shows a sample response to an input of WD.

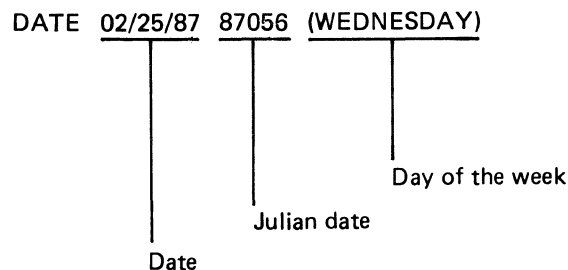


Figure 2-252. WD Display

Details

The current date is based on the date entered by the operator at system initialization time or through the DR (Date Reset) command.

The space between WD and the minus sign is required, but not between WD and the plus sign (+).

Examples

WD + 344

displays the date 344 days in the future.

WD - 2

displays the date of the day before yesterday.

WD 75360

displays the date and day of the week of the 360th day in 1975.

See also:

DR (Date Reset); TR (Time Reset); and WT (Display Time) commands.

The DATE and TIME records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

WJ (DISPLAY WAITING JOBS)

Keyboard Input Message

Access Level: 0

WJ lists only those jobs that are currently waiting. You can list a subset of the waiting jobs by using syntax options of the command.

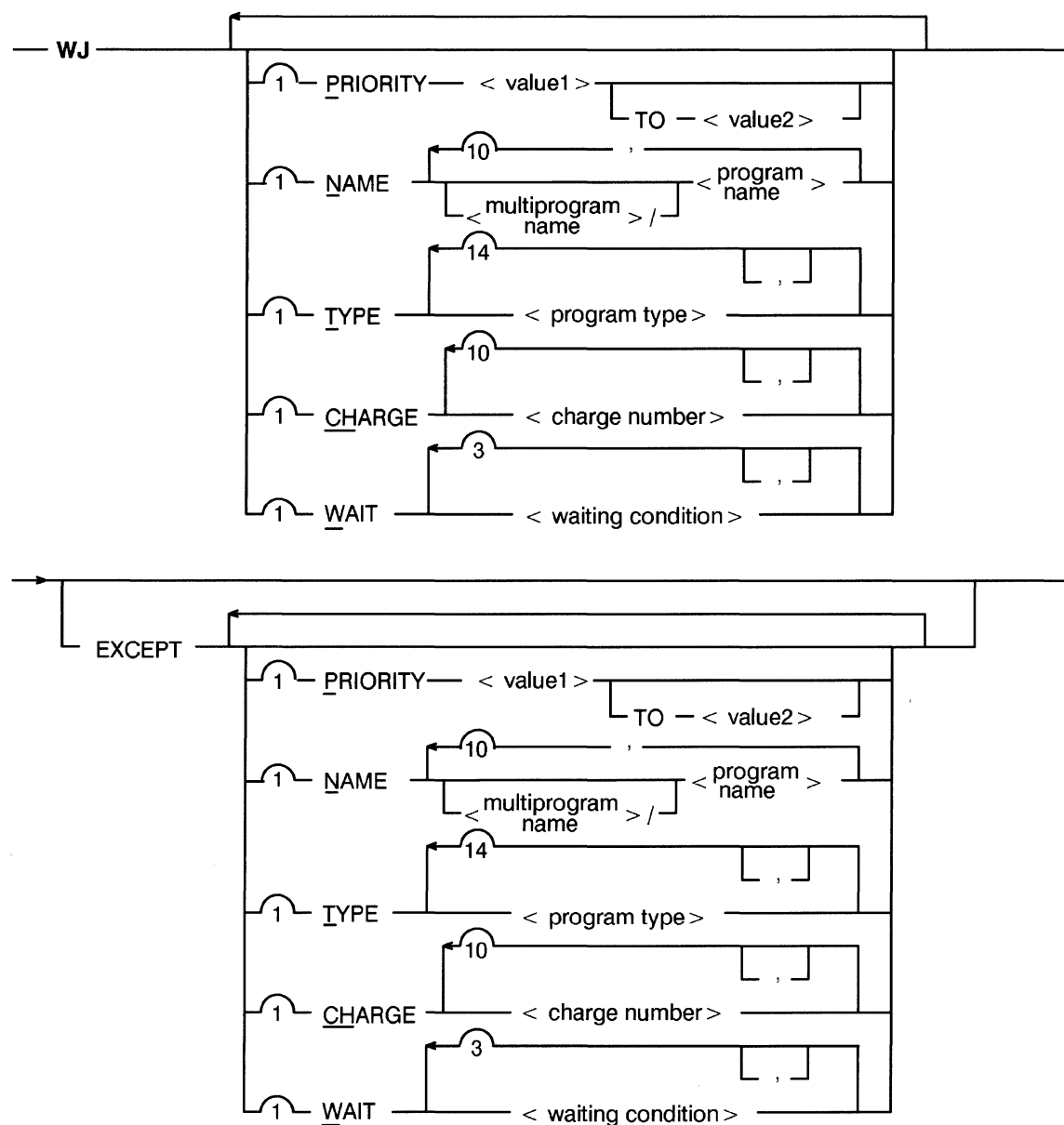


Figure 2-253. WJ Command Syntax

■ **PRIORITY** <value1> TO <value2>

Selects jobs based on processor priority. You have the option of selecting a single priority or a range of priorities. <value1> and the optional <value2> must be integers between 1 to 9. <value1> must be less than or equal to <value2>. The abbreviation P is accepted for PRIORITY.

■ **NAME** <multiprogram name> / <program name>

Selects jobs based on the program name and (optionally) on the multiprogram name. You can enter a maximum of 10 name selections, separated by commas. The abbreviation N is accepted for NAME.

You can also select jobs based on part of a name by masking the name using the equal sign as a masking character. Use one equal sign for each unknown character position in the name. For example,

WJ NAME DMP=== EXCEPT NAME DMPOUT

displays all programs in the mix beginning with the three characters DMP, but does not display the program DMPOUT.

■ **TYPE** <program type>

Selects jobs based on the type of program. The abbreviation T is accepted for TYPE. You can enter a maximum of 14 type selections. The available program types are:

EXECUTE	Program is executing
COMPILER	Program generates object code
DMPALL	Program is DMPALL intrinsic
LOADMP	Program is LOADMP intrinsic
PACKUP	Program is PACKUP intrinsic
DSKOUT	Program is DSKOUT intrinsic or diskpack SQUASH intrinsic
MCS	Program is Message Control System
TIMESHARE	Program is executed as time sharing task
SHAREHNDL	Program is a time sharing handler program (for example, CANDE)
HANDLERS	Program is any type of handler (BNA, WFL, time sharing)
COMPCHAR	Program that generates object code, executed as a timesharing task
DMSPROG	Program is a DMSII control program (for example, DBP)
WFLHNDL	Program is a Work Flow Language (WFL) handler
BNAHNDL	Program is a BNA network architecture handler
COPY	Program is SYSTEM/COPY intrinsic
SORT	Program is SORT: or SORT. intrinsic
UTILITY	Program is any type of intrinsic (DMPALL, LOADMP, PACKUP, DSKOUT, SYSTEM/COPY)

CHARGE <charge number>

Selects jobs based on the charge number the job is executed under. You can enter as many as 10 charge number selections and 10 charge number exceptions. The abbreviation CH is accepted for CHARGE.

You can also select jobs based on part of a charge number by masking the charge number using the equal sign as a masking character. Use one equal sign for each unknown character position. For example,

WJ CH 66===== EXCEPT CH 660001, 669999

displays jobs executed under a charge number beginning with the characters 66, but does not display jobs executed under charge numbers 660001 or 669999.

WAIT <waiting condition>

Selects jobs that are waiting for a specific condition to be satisfied (for example, waiting hardware, waiting memory and so forth). You can enter as many as 3 waiting conditions and 3 waiting condition exceptions. The abbreviation W is accepted for WAIT.

The available types of waiting conditions are:

<u>SPACE</u>	Waiting for memory, disk space or diskpack space
<u>DEVICE</u>	Waiting for hardware device to become available
<u>OPERATOR</u>	Waiting for operator to perform an action
<u>EVENT</u>	Waiting for a software event to be signalled (for example, STOQUE, CRCR and so forth)

EXCEPT

All text following the keyword EXCEPT is treated as exceptions to the selections entered previously. All of the exceptions must be grouped together after the keyword EXCEPT; you cannot enter a selection and an exception followed by another selection.

Details

The WJ command corresponds to the AD command <table> element option WM for ODT display. WJ and WM display the jobs waiting in the mix, including the processor and memory priorities and memory required for each job.

Sample Displays

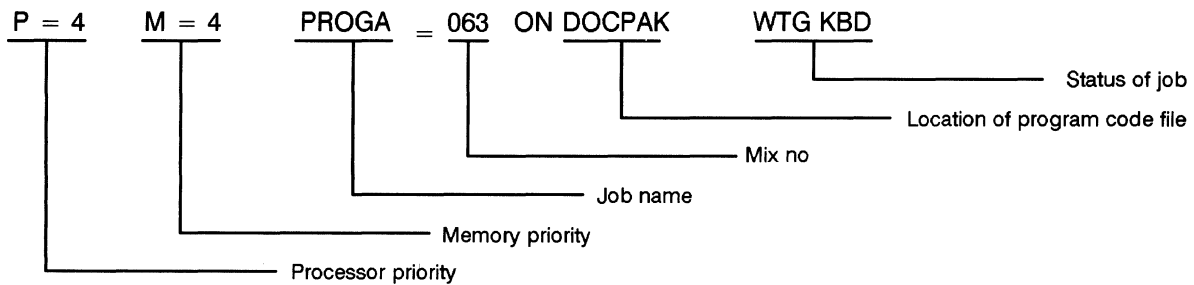


Figure 2-254. WJ Display

NULL MIX

No jobs are waiting.

Reasons for Waiting

Some reasons a job can be waiting:

- There is not enough memory for it.
- It is waiting for a peripheral device to become available.
- It is waiting to receive a message from another program.
- It is waiting for the operator to give it a message with the AX (ACCEPT) command.
- It has received no-file or duplicate-file message.

See Also:

AD (Specify ODT/OCS Screen Parameters); AJ (Display Active Jobs); MX (Display Active Mix); and WS (Display Schedule) commands.

WM (Display MCP Version)

Keyboard Input Message

Access Level: 0

The WM command displays the name, release level, patch level, and version date of the MCP currently running; a list of system options that are set; the system number; the hostname; and the amount (in KD) of main memory and MCP QWIK memory.

WM

Figure 2-255. WM Command Syntax

Sample Response

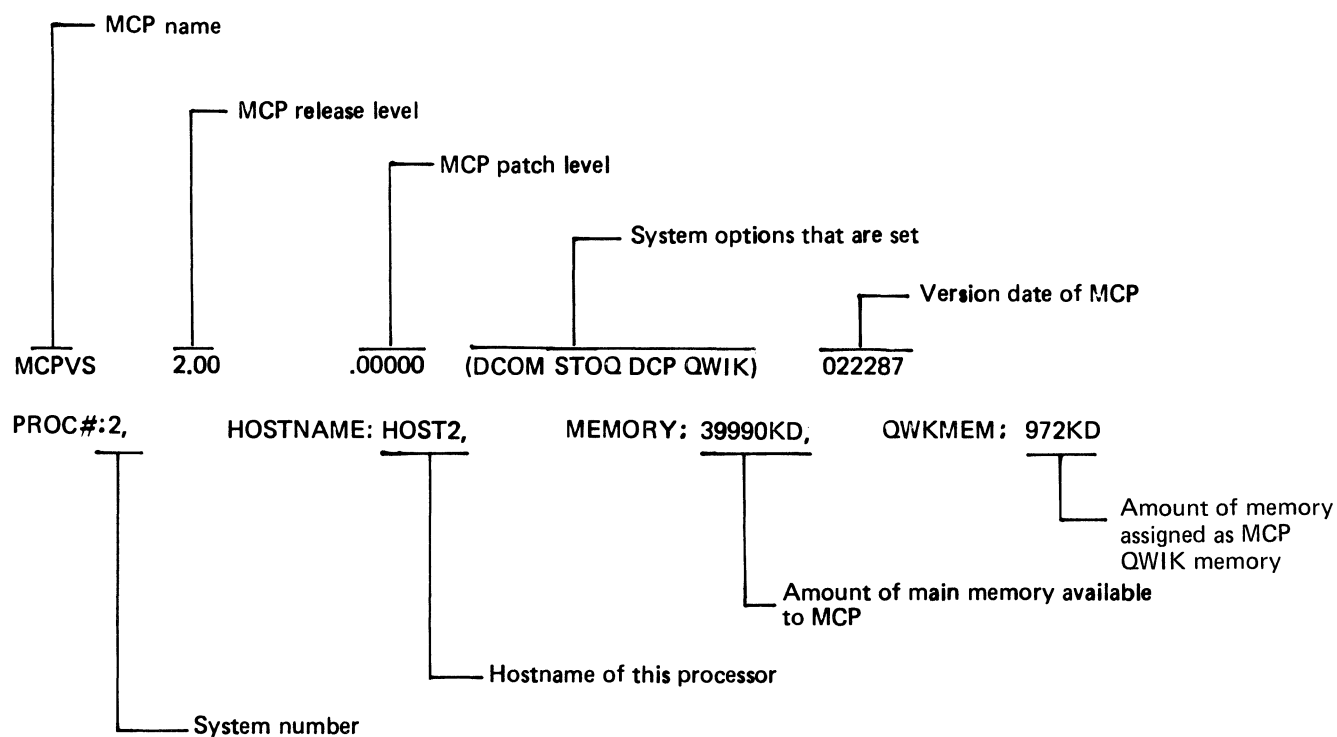


Figure 2-256. WM Display

V Series MCP/VS System Software Operations Guide, Volume 2
The System Commands

Explanation of Response

System options are set by USE records in the system configuration file or by the SO (Set System Options) command. They can be reset by the RO (Reset System Options) command.

The version date is in mmddyy format.

The hard-wired processor number is set by the technician when the processor is installed.

The hostname is set by the HOSTNAME record in the system configuration file or the HN (Identify Or List Host Name) command. It identifies this processor in communications, the system logs, and networking environments.

MCP QWIK memory is memory available to store the most recently used MCP overlays, so that they will not have to be accessed from disk if they are needed again. This feature increases performance and is set by the LIMIT QWKMEM record in the system configuration file.

See also:

The HN (Identify Or List Host Name); RO (Reset System Options); and SO (Set System Options) commands.

The HOSTNAME, LIMIT QWKMEM, and USE records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

Access Level: 0

— WO

Figure 2-257. WO Command Syntax

The User QWIK pool is memory dedicated to storing the most recently used user program overlays, so that they will not have to be accessed from disk if they are needed again. This feature usually increases performance. It can be set by the USE QWIK record in the system configuration file or by the SO (Set System Options) command. It can be reset by the RO (Reset System Options) command.

Sample Response

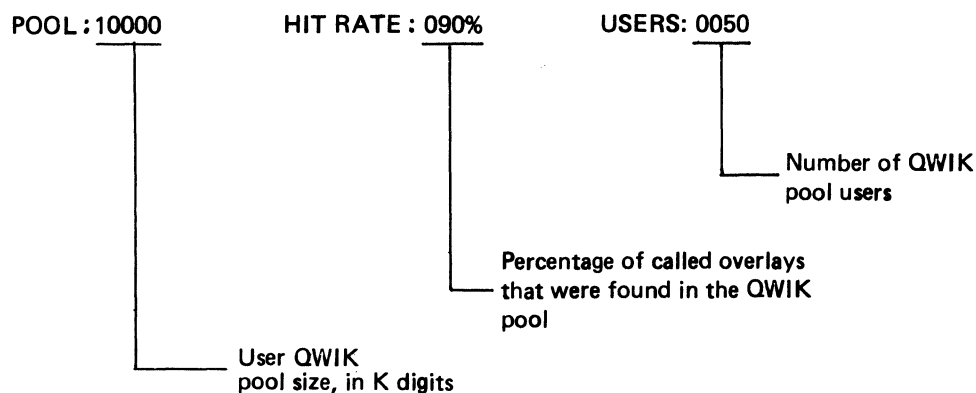


Figure 2-258. WO Display

Explanation of Response

Pool Size

is the size of the QWIK pool set by the LIMIT QWIKPOOL record in the system configuration file or the QWIKPOOL (Set Program Overlay Buffer Pool) command.

Hit Rate

is the percentage of times that an overlay was called and found in the QWIK pool, therefore it did not have to be retrieved from disk.

USERS

is the number of tasks currently retrieving overlays from the QWIK pool.

Details

Statistics, tabulated by QWIK for each five-minute interval, are placed in the Run log and can be interpreted by RLGOUT or the Time Analysis and Billing System II (TABSII). Among these statistics are the following:

- Number of times an overlay was found in the pool.
- Number of times an overlay was called.
- The percentage of overlay calls in which the overlay was found in the QWIKMEM pool

More detailed statistics on user QWIK memory are available through the Run log report program and the TABSII program product. See the *V Series System Software Logging Operations Reference Manual* and the *B 2000/B 3000/B 4000/V Series TABS II Installation and Operations Guide*.

See Also:

QWIKPOOL (Set Program Overlay Buffer Pool); SO (Set System Options); RO (Reset System Options); and TO (Type System Options) commands.

The USE QWIK and LIMIT QWIKPOOL records in the system configuration file, in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

The *B 2000/B 3000/B 4000/V Series TABS II Installation and Operations Guide*.

WQ (Display STOQUE Count)

Keyboard Input Message

Access Level: 0

The WQ command displays the number of items in a storage queue (STOQ).

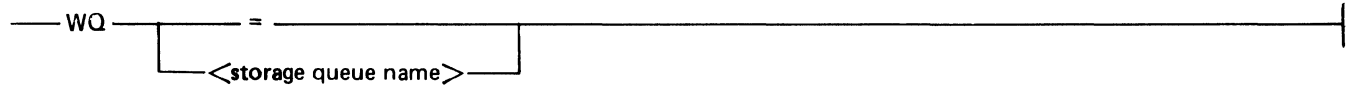


Figure 2-259. WQ Command Syntax

=
displays the names and item counts of all storage queues.

<storage queue name>
is the name of the storage queue. This name is assigned by the programs using the queue.

Explanation of Storage Queue (STOQ)

A storage queue (STOQ or STOQUE) is a buffer in memory used to transfer information asynchronously from one program to another. This means that if a program sends a message to another program through STOQ and that program is not ready to accept the message, the message is stored in the storage queue until the other program is ready to retrieve it.

The receiving program need not be in the mix when the message is sent. The receiving program retrieves its messages, one at a time.

A storage queue is created when a program puts a message in it or requests a message from it. It continues to exist as long as there are messages in it, or as long as a program is awaiting a message from it.

Each storage queue has a name, assigned by the programs according to the rules of the programming language. The sending program and the receiving program must agree on this name.

Examples

WQ JACK

displays the number of messages in the storage queue named JACK.

WQ =

displays the names and the number of messages waiting in all active storage queues.

Sample Responses

QUEUEA 5

There are five messages waiting in a storage queue named QUEUEA.

QUEUEB 0 (WTG)

There are no messages in the storage queue named QUEUEB, but a program has issued a request for a message from this queue. The program is waiting for the message.

See Also:

RQ (Remove Storage Queue Entry) command.

The USE STOQ record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

Complete details on storage queues are given in the various language manuals and in the *V Series Programming Reference Manual*.

WS (Display Jobs in Schedule)

Keyboard Input Message

Access Level: 0

The WS command displays a list of jobs that are scheduled to execute but are not yet executing.

—WS—

Figure 2-260. WS Command Syntax

Details

If no jobs are in the schedule, the message "NULL SCHED" appears on the ODT.

Jobs can be removed from the schedule by the RS (Remove Job From Schedule) command.

See Also:

AFTER (Hold Program); RA (Remove After-Linkage); RS (Remove Job From Schedule); WJ (Display Waiting Jobs); and WY (Display Job Status) commands.

WT (Display Time)

Keyboard Input Message

Access Level: 0

The WT command displays the time of day. The time is given in 24-hour format (military time). The time of day used by the system is set at system initialization and can be changed by the operator through the TR (Time Reset) command.

—WT—

Figure 2-261. WT Command Syntax

Sample Response

TIME 13:46:26
indicates that the time is 1:46 pm and 26 seconds.

See Also:

DR (Date Reset); TR (Time Reset); and WD (Display Date) commands.

The DATE and TIME records in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

WX (DISPLAY REMOVED SPACE)

Keyboard Input Message

Access Level: 8

The WX command lists all areas of memory, disk, or pack that have been removed from the system by the operator.

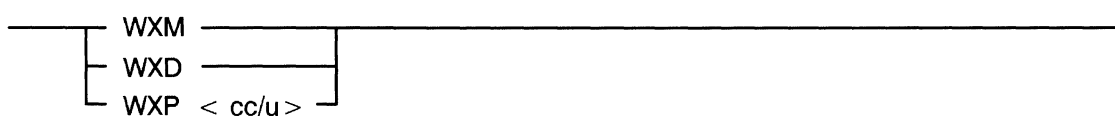


Figure 2-262. WX Command Syntax

WXM

displays deleted areas of memory.

WXD

displays deleted areas of disk.

WXP <cc/u>

displays deleted areas of pack. The <cc/u> is the channel number/unit number of the pack drive.

Details

Areas of memory, disk, or pack can be removed from the system if they become faulty. The operator does this with the XM (Remove Memory From System), XD (Remove Disk Segments), or XP (Remove Pack Sectors) command.

Areas that have been removed with the XM, XD, or XP command can be restored to the system with the RXM, RXD, RXP (Return Deleted Space) commands.

Examples

WXM

displays the areas of memory that have been deleted.

WXD

displays the disk areas that have been deleted.

WXP 8/3

displays which areas have been deleted from the pack mounted on the drive that is on channel 8/unit 3.

See Also:

RX (Return Deleted Space); XD (Remove Disk Segments); XM (Remove Memory From System); and XP (Remove Pack Sectors) commands.

WY (DISPLAY JOB STATUS)

Keyboard Input Message

Access Level: 0

The WY command displays the status of a job and the job's associated messages. WY also displays the status of all jobs in the mix. You can display a subset of jobs in the mix by using syntax options of the command.

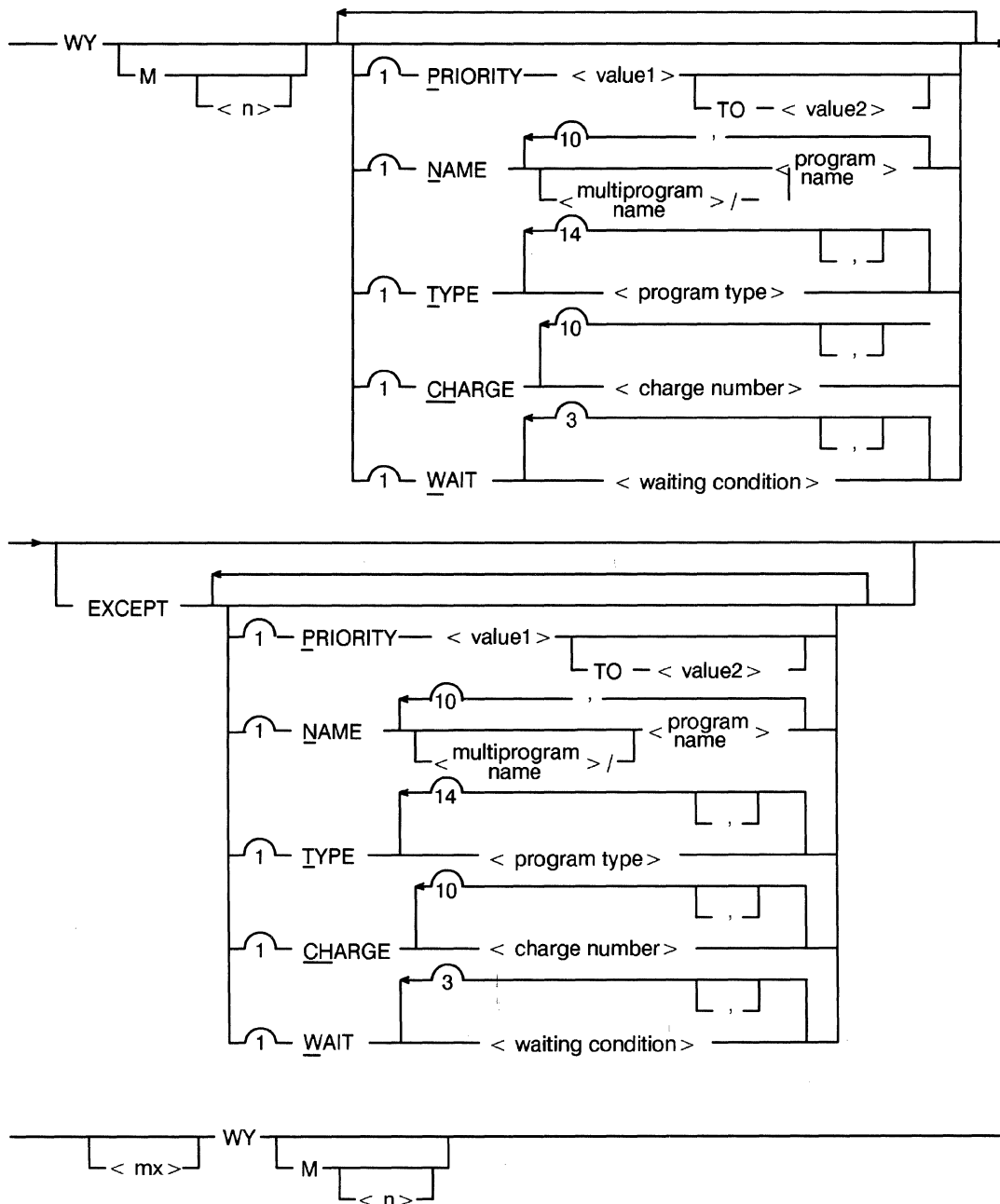


Figure 2-263 WY Command Syntax

C <mx>

the mix number of a job on the system. The operating system displays the status of the job associated with the mix number. <mx> cannot be 0 or 1.

Because this option selects one specific job, the job selection options (PRIORITY, NAME, and so forth) are ignored when this option is used.

M

causes the MCP to display both previous ACCEPT and DISPLAY messages. For example, <mx> WY M displays:

- the status of the job.
- messages sent to the job through the AX (Response To An ACCEPT) command.
- messages sent from the job for display on the ODT (for example, by the DISPLAY verb in COBOL).

<n>

is the number of messages to display and can be a number from 1 to 10. The default is 2.

PRIORITY <value1> TO <value2>

Selects jobs based on processor priority. You have the option of selecting a single priority or a range of priorities. <value1> and the optional <value2> must be integers between 1 to 9. <value1> must be less than or equal to <value2>. The abbreviation P is accepted for PRIORITY.

C NAME <multiprogram name> / <program name>

Selects jobs based on the program name and (optionally) on the multiprogram name. You can enter a maximum of 10 name selections, separated by commas. The abbreviation N is accepted for NAME.

You can also select jobs based on part of a name by masking the name using the equal sign as a masking character. Use one equal sign for each unknown character position in the name. For example,

WY NAME DMP=== EXCEPT NAME DMPOUT

displays all programs in the mix beginning with the three characters DMP, but does not display the program DMPOUT.

■ TYPE <program type>

Selects jobs based on the type of program. The abbreviation T is accepted for TYPE. You can enter a maximum of 14 type selections. The available program types are:

EXECUTE	Program is executing
COMPILER	Program generates object code
DMPALL	Program is DMPALL intrinsic
LOADMP	Program is LOADMP intrinsic
PACKUP	Program is PACKUP intrinsic
DSKOUT	Program is DSKOUT intrinsic or diskpack SQUASH intrinsic
MCS	Program is Message Control System
TIMESHARE	Program is executed as time sharing task
SHAREHNDL	Program is a time sharing handler program (for example, CANDE)
HANDLERS	Program is any type of handler (BNA, WFL, time sharing)
COMPSHAR	Program that generates object code, executed as a timesharing task
DMSPROG	Program is a DMSII control program (for example, DBP)
WFLHNDL	Program is a Work Flow Language (WFL) handler
BNAHNDL	Program is a BNA network architecture handler
COPY	Program is SYSTEM/COPY intrinsic
SORT	Program is SORT: or SORT. intrinsic
UTILITY	Program is any type of intrinsic (DMPALL, LOADMP, PACKUP, DSKOUT, SYSTEM/COPY)

■ CHARGE <charge number>

Selects jobs based on the charge number the job is executed under. You can enter as many as 10 charge number selections and 10 charge number exceptions. The abbreviation CH is accepted for CHARGE.

You can also select jobs based on part of a charge number by using an equal sign as a masking character. Use one equal sign for each unknown character position. For example,

WY CH 66===== EXCEPT CH 660001, 669999

displays jobs executed under a charge number beginning with the characters 66, but does not display jobs executed under charge numbers 660001 or 669999.

■ WAIT <waiting condition>

Selects jobs that are waiting for a specific condition to be satisfied (for example, waiting hardware, waiting memory and so forth). You can enter as many as 3 waiting conditions and 3 waiting condition exceptions. The abbreviation W is accepted for WAIT.

The available types of waiting conditions are:

SPACE	Waiting for memory, disk space or diskpack space
DEVICE	Waiting for hardware device to become available
OPERATOR	Waiting for operator to perform an action
EVENT	Waiting for a software event to be signalled (for example, STOQUE, CRCR and so forth)

EXCEPT

All text following the keyword EXCEPT is treated as exceptions to the selections entered previously. All of the exceptions must be grouped together after the keyword EXCEPT; you cannot enter a selection and an exception followed by another selection.

Job Status

Some possible job statuses include:

EXECUTING

The program is being processed.

WAITING

The job is waiting for something to happen before it can continue processing. It can be waiting for a message from another program, or for a message from the operator, or for a message from a data communication device, or for a peripheral device to become available, or for additional space in memory.

TERMINATING

The MCP is going through the routines for ending this job.

Examples

WY

displays the status of all jobs in the mix.

2 WY

displays the status of the job associated with mix number 002.

WY PR 1-6 CH 661 === EXCEPT CH 661000

displays the status of all jobs with processor priorities from 1 through 6 that are executing under a charge number beginning with 661, except for any jobs executed under a charge number of 661000.

22 WY M 10

displays the status of the job associated with mix number 022; the last 10 messages that have been sent to that job through the AX command; and/or the last 10 messages that job has displayed on the ODT.

See also:

AJ (Display Active Jobs); WJ (Display Waiting Jobs); and WS (Display Schedule).

XA (ADD/REMOVE A CHANNEL FROM AN EXCHANGE)

Keyboard Input Message

Access Level: 8

The XA command can be used to add or remove a channel from an exchange.

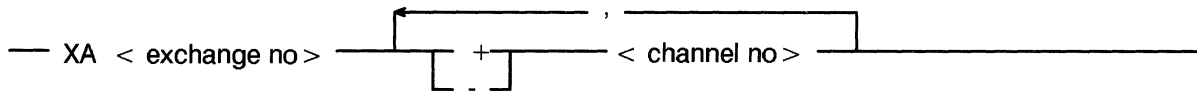


Figure 2-264. XA Command Syntax

<exchange no>

indicates the exchange to which you are adding or from which you are deleting a channel.

If you enter an exchange number that has not been previously declared in an EXCHANGE record in the system configuration file, the system creates an exchange with the exchange number as the primary channel, and the channel you were adding as the single alternate channel.

<channel no>

indicates the channel you are adding or deleting. This channel cannot be a member of another exchange, and must be of a hardware type identical to the primary channel of the exchange.

Examples

XA 11 + 04

adds channel 04 to exchange 11.

DLP 24 DISK HSTLUB

DLP 25 DISK HSTLUB

DLP 50 DISK HSTLUB

■ XA 24 + 25, + 50

adds DLPs with channel numbers 24, 25, and 50 to the system environment. Channel number 24 becomes the primary channel and exchange. Channels 25 and 50 are added as alternate channels.

See also:

DD (Delete DLP); DLP (Add DLPs); and SHOW (Display System Information) commands.

The EXCHANGE record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

XC (Inhibit/Uninhibit Or Display Status Of A Channel)

Keyboard Input Message

Access Level: 8

The XC command inhibits, uninhibits, or displays the status of a channel.

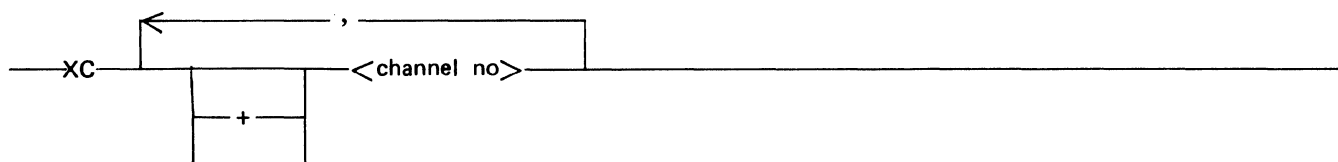


Figure 2-265. XC Command Syntax

XC <channel no>
displays the status of the channel. Status can be INHIBITED or NOT INHIB.

XC + <channel no>
inhibits the specified channel, making it unavailable to the system.

XC - <channel no>
uninhibits the specified channel, making it available to the system.

Details

The XC command is generally used to uninhibit a channel after an unrecoverable channel error has inhibited it.

Inhibited or uninhibited status is retained across halt/loads.

Restrictions

The XC command cannot be used to inhibit the HLSPO or the primary disk channel.

Examples

XC +5
inhibits channel number 5, making it unavailable to the system.

XC -4
uninhibits channel number 4, making it available to the system.

XC 30
displays the availability status of channel number 30.

See Also:

XA (Add/Remove a Channel From An Exchange) command.

XD (Remove Disk Segments)

Keyboard Input Message

Access Level: 8

The XD command removes disk segments from the system.

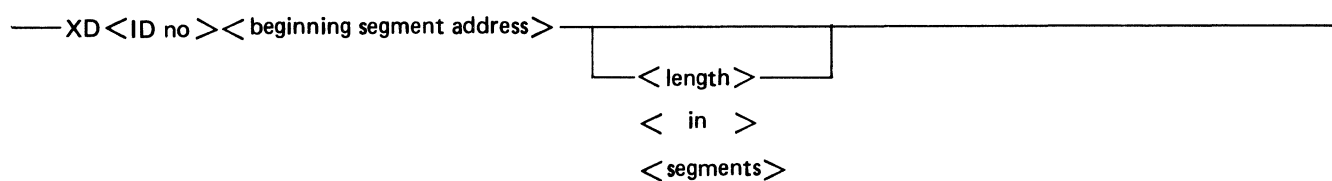


Figure 2-266. XD Command Syntax

<ID no>

is the ID number of the disk unit to be affected.

<begin segment address>

is the segment address of the first segment to be removed.

<length>

is the number of segments to be removed. The default is 1,000 segments.

Restriction

A disk area that is assigned to a file cannot be removed by the XD command. Use the KS (Analyze Disk Space) command to ascertain if a disk area is assigned to a file.

Details

The XD command removes the specified segments from the Disk Available table of the MCP, making them unavailable.

Disk segments are returned to the system by the RXD (Return Deleted Space) command.

Example

XD 10 10000 500

deletes segments 10000 through 10499, a total of 500 segments, from disk unit 10.

See Also:

KS (Analyze Disk Space); RXD (Return Deleted Space); XM (Remove Memory From System); XP (Remove Pack Sectors); and WX (Display Removed Space) commands.

XM (Remove Memory From System)

Keyboard Input Message

Access Level: 8

The XM command removes a specified area of memory from the system.

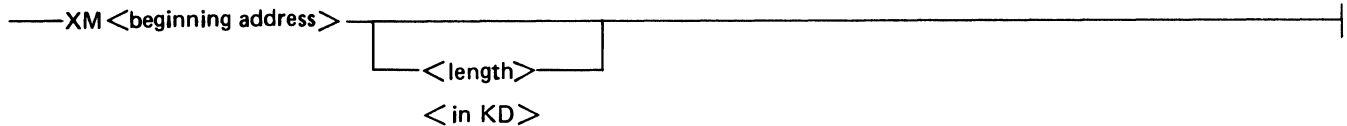


Figure 2-267. XM Command Syntax

<beginning address>

is the absolute address in K digits of the first location to be removed. If <length> is not specified, 1 KD of memory is removed.

<length>

is the number of memory locations in K digits to be removed. The default is 1 KD.

Details

The memory cannot be in use.

A halt/load subsequent to the XM command does not return the space to the system unless the memory map has been destroyed.

Memory is returned to the system by a cold start or an RXM command.

Example

XM 310 5

removes 5 KD of memory, starting at absolute address 310000.

See Also:

RX (Return Deleted Space); WC (Display Available Memory); and WX (Display Removed Space) commands.

XP (Remove Pack Sectors)

Keyboard Input Message

Access Level: 8

The XP command removes pack sectors from the system.

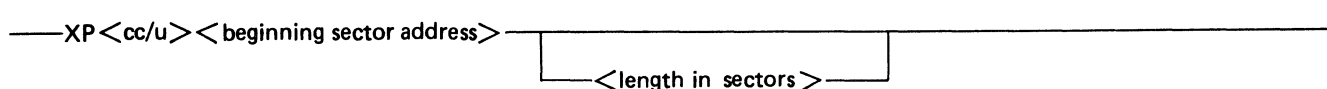


Figure 2-268. XP Command Syntax

<cc/u>

is the channel number/unit number of the pack drive to be affected.

<beginning sector address>

is the location in decimal of the first sector to be removed. The limits for <beginning sector address> are 1 through 16700000 or subject to drive limitations. If <length in sectors> is not specified, 1000 sectors are removed.

<length in sectors>

is the number of sectors in decimal to be removed. The limits for <length in sectors> are 1 through 16700000 or subject to drive limitations. Default is 1000 sectors.

Restrictions

The pack must not be in use.

An area that is assigned to a file cannot be removed by the XP command.

Details

The XP command removes the pack sectors from the master available table of the diskpack, making these sectors unavailable for assignment. The PS (Analyze Pack Space) command can be used to ascertain unavailable sectors.

Diskpack sectors are returned to the system by an RXP command.

Examples

XP 3/12 2000 1500

deletes sectors 2000 through 3499, a total of 1500 sectors, from the diskpack that is mounted on the drive that is declared as unit 12 on channel 3.

See Also:

PS (Analyze Pack Space); RX (Remove Deleted Space) and WX (Display Removed Space) commands.

APPENDIX A

USING RAILROAD SYNTAX DIAGRAMS

A railroad diagram is a way of graphically representing the syntax of a command or statement. The graphic representation shows you if items are required or optional, whether they are constants or variables, the order in which they should appear, how often you can repeat them, and if any specific punctuation is required.

Railroad diagrams use a variety of symbols to convey this information. You should keep in mind, though, that some of the symbols contained in the railroad diagram are not part of the actual command or statement as you would enter it at the keyboard. For example, a railroad diagram depicting the command to save a file appears as:

— SAVE <file-name> —|

The word, file-name, is enclosed in angle brackets (< >) only to indicate that it is a variable item, which will be discussed in detail below. When you enter this command at your terminal, you would not type in the angle brackets. You would simply enter the appropriate file name for the file you were saving. The actual command would look like this:

SAVE MYFILE

In this example, MYFILE is the name of the file being saved.

Additionally, railroad diagrams that depict specific commands (such as the Save command above) normally do not include a depiction of a TRANSMIT key. However, to execute a given command, you might have to press a TRANSMIT key (or key with a different name that performs a transmit function) after you have entered the command information.

READING A RAILROAD DIAGRAM

Generally, you read a railroad diagram from left to right, top to bottom, just as you read text in a book. Sometimes, though, there are exceptions. In these cases, arrows are used to show you the correct direction to read the railroad diagram.

The end of a railroad diagram is denoted by a vertical bar (|):

— SAVE <file-name> —|

If a diagram is too long to fit on one line of a page, a right angle bracket (>) is placed at the end of the first line and another at the beginning of the next line. For example, the LIST command shown below could be divided as follows:

— LIST —|
 | RECORDS | <range> | PAGE |

ITEMS OF A RAILROAD DIAGRAM

There are a variety of items that can make up a railroad diagram. Typically, not all possible items are used in each command, so not all diagrams include all the possible items. Below is a list of items that can be used in railroad diagrams:

- Constants
- Variables
- Required items
- Optional items
- Loops and Bridges
- Punctuation

Constants

Constants are items that must be entered exactly as they are depicted in the railroad diagram. They cannot be modified. You should be able to easily recognize constants in railroad diagrams because they are always UPPER CASE and are not enclosed in angle brackets.

Often, part of the constant will be underlined. The underlining below the letters of the constant indicates the minimum characters you may enter to form the particular command. For example:

— SAVE <file-name> —

In this example, you could enter just the letters SA to perform the SAVE operation, followed by a file name, or you could enter the whole command, SAVE, followed by a file name. If no underlining is present, you must enter all the letters of the constant.

Variables

Variables are items that you replace with appropriate data to suit your particular situation. Variables appear in a railroad diagram enclosed in right and left brackets (< >). For example:

— SAVE <file-name> —

The word file-name is a variable. When you actually use the SAVE command, you will enter your own file name for the data you are saving (without the angle brackets). For example, you could enter:

SA MYFILE or SAVE XYZ

In these two examples, the first file name is MYFILE and the second file name is XYZ. SA is acceptable because it is underlined in the diagram.

Required Items

Required items are items that must be entered in the command or statement you are forming. You cannot omit them. Required items are any items that appear directly on the horizontal line of the railroad diagram:

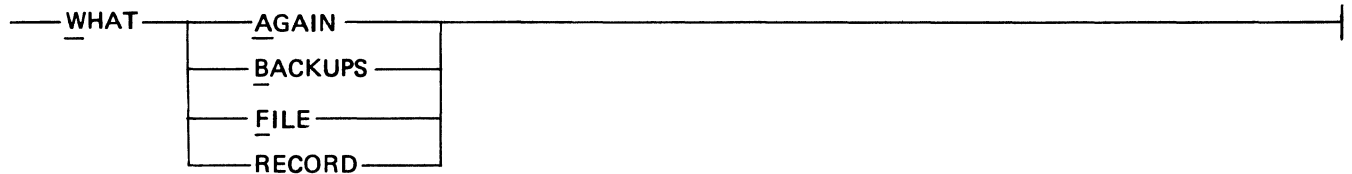
—SAVE <file-name>—

SAVE above is a constant, and <file-name> is a variable. But both items are required since they appear directly on the horizontal line.

Optional Items

Optional items appear in a vertical list (one below another) within a railroad diagram. There are two types of optional items. Those that are required and those that are not required. Both types are called "optional" because you can select one item from a list of items to use in the command you are forming.

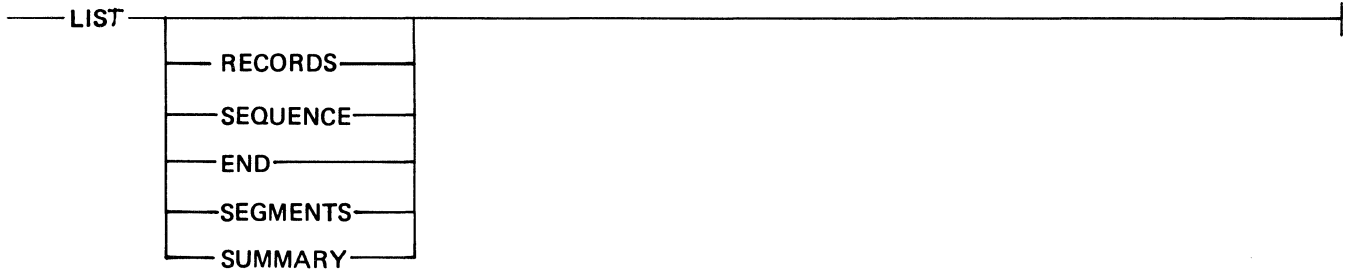
You can tell if an option is required or not by the position of the first option in the list of items. If the first option in the list appears directly on the horizontal line then the item is a required option. The WHAT command below includes a required option, which can be selected from the four listed items. That is, you must select one of the optional items to complete the command structure correctly.



In this sample WHAT command, you must select one of the four optional items, AGAIN, BACKUPS, FILE, RECORD. Notice also that since the underlining allows single letter entries for both WHAT and all the options, if you selected FILE, you could enter:

WHAT FILE or W FILE or WHAT F or even W F

An option that is not required is indicated when the first option in the list appears below the horizontal line (leaving the line unbroken). In this case, none of the options is actually required to complete the command structure. For example:

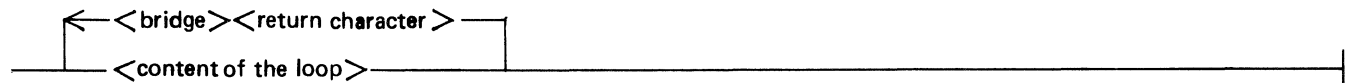


In this sample LIST command, you may or may not enter one of the options since the first option appears below the horizontal line. To complete this sample List command correctly, you could enter just the word LIST. Other possible combinations would be: LIST RECORDS, LIST SEQUENCE, LIST END, LIST SEGMENTS, and LIST SUMMARY.

Loops And Bridges

A loop is an item or group of items that you can repeat. The number of repetitions allowed is controlled by an item called the bridge.

A loop can span all or part of a railroad diagram. It always consists of at least two horizontal lines, one below the other, as in the following example:



The top line is a right-to-left path that contains information (the bridge and return character) about repeating the loop.

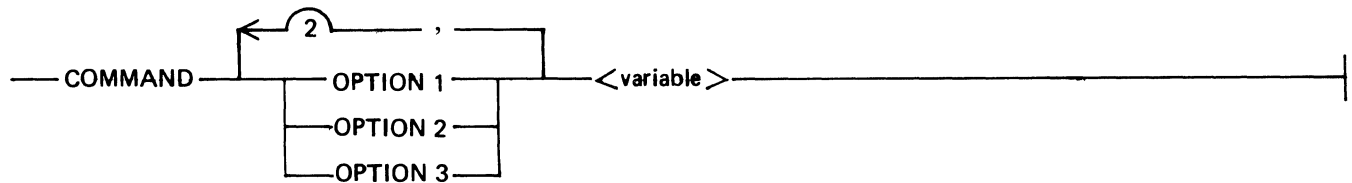
The bridge shows the maximum number of times you can go through the right-to-left path (in other words, repeat the loop). The bridge is an integer enclosed in sloping lines, / \, for example, /4\. Not all loops have bridges. Those that do not can be repeated any number of times.

The return character is the character to use before each repetition of the loop (often, a comma). Not all loops contain a return character; if none is shown, just enter one or more spaces before repeating the loop.

The other lines show the content of the loop (the data you enter each time you go through the loop). This can be any combination of optional items, required items, and even other loops. The content of a loop ranges from the simple (one item) to the very complex (many items, lists, and loops).

Example 1. A Simple Loop

The structure of a simple loop is shown below in a generic command example:



In the generic command illustrated above, the loop is on top of OPTION 1. But loops can span all or any part of a railroad diagram, either above or below the main horizontal line.

The number of repetitions you are allowed within a loop is indicated by the bridge.

In the sample loop above, the bridge /2\ and the punctuation (,) indicate the number of times the you can go through the loop and what punctuation must be included in the command structure. The comma is used here as a return character before each repetition of the loop.

Based on the sample loop structure above, you could enter a number of valid combinations for the command. Three such allowable command structures you might enter are:

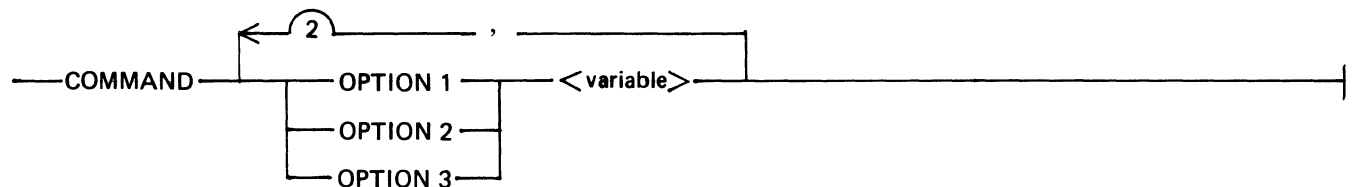
COMMAND OPTION 1, OPTION 2 variable

COMMAND OPTION 1, OPTION 3, variable

COMMAND OPTION 1, OPTION 2, OPTION 3 variable

Example 2. A More Complex Loop

Below is a sample diagram where the loop includes a <variable>. With this structure, you can specify a different variable for each option, complicating the loop itself:



Based on this sample loop structure, you could enter a number of valid combinations for the command. Three such allowable command structures you might enter that include the <variable> are:

COMMAND OPTION 1 <variable>, OPTION 2 <variable>

COMMAND OPTION 2 <variable>, OPTION 1 <variable>, OPTION 3 <variable>

COMMAND OPTION 3 <variable>, OPTION 2 <variable>, OPTION 1 <variable>

Punctuation

Punctuation is an item that is used as a delimiter, or boundary, between other items in the command itself. In the discussion of "Loops and Bridges" above, the comma was used as required punctuation to set a boundary between items within the loop. In some cases, command punctuation may be replaced by spaces.

CONCLUSION

Railroad diagrams are used extensively in this manual. Some of the actual diagrams are considerably more complex than those presented in the preceding paragraphs. Nevertheless, the principles that guide the logic of the diagrams remain the same. Obviously, the more you work with railroad diagrams, the more easily you will be able to interpret even the most complex diagrams.

This appendix contains the syntax of those commands that initiate the **LOADMP**, **PACKUP**, **PBDOUT**, and **PBTOUT** utilities. These commands will become unavailable at a later release.

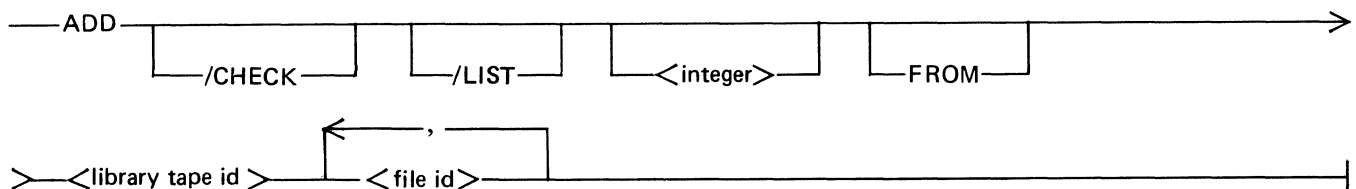
LOADMP copies files to and from, checks files from, and moves files to and from disk. PACKUP performs similar functions on pack.



ADD (Copy Files From Library Tape To Disk or Pack)
CHECK (Verify Library Tape),
DUMP (Copy Files To Library Tape)
LOAD (Move Files From Library Tape to Disk or Pack)
UNLOAD (Move Library Files Between Media, Remove Old Files)

Figures B-1 through B-7 show ADD (Copy Files From Library Tape To Disk or Pack) command syntax. The various options show ADD command facilities for pack users to ADD files to disk, pack, and magnetic tape.

The syntax shown in Figure B-1 invokes LOADMP to copy files from a library tape to disk.



C

ADD Option 1:

Option 1 invokes PACKUP to transfer files from pack to disk that are not already present.

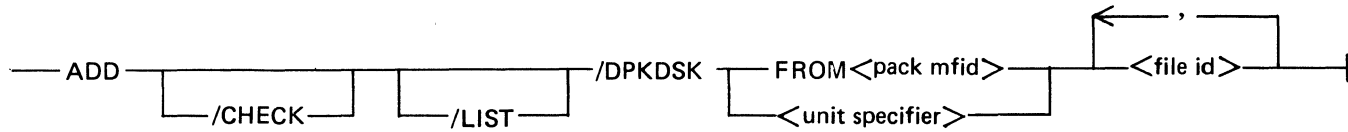


Figure B-2. ADD Command Syntax (PACK to DISK)

ADD Option 2:

Option 2 invokes PACKUP to transfer files from a pack library tape to pack that are not already present.

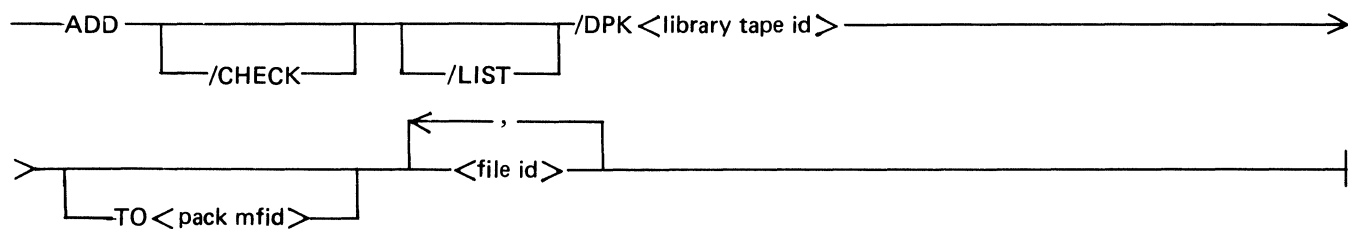


Figure B-3. ADD Command Syntax (PACK Tape to PACK)

ADD Option 3:

Option 3 invokes PACKUP to transfer files from disk to pack that are not already present.

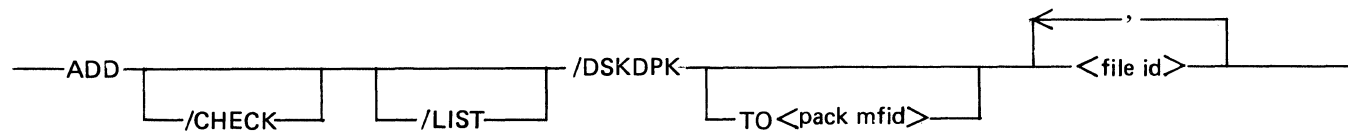


Figure B-4. ADD Command Syntax (DISK to PACK)

ADD Option 4:

Option 4 invokes PACKUP to transfer files from pack to pack that are not already present.

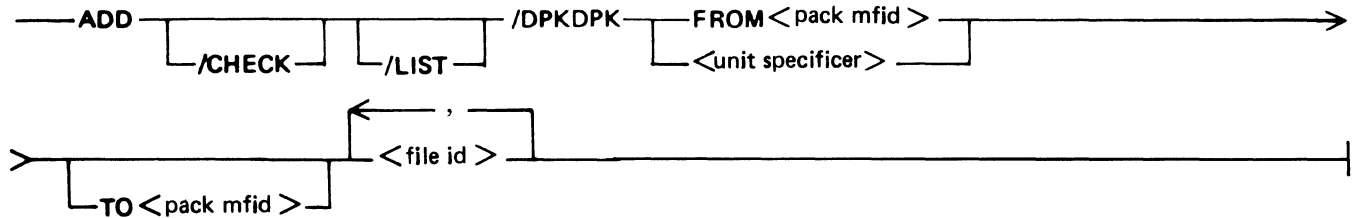


Figure B-5. ADD Command Syntax (PACK to PACK)

ADD Option 5:

Option 5 invokes PACKUP to transfer files from a disk library tape to pack that are not already present.

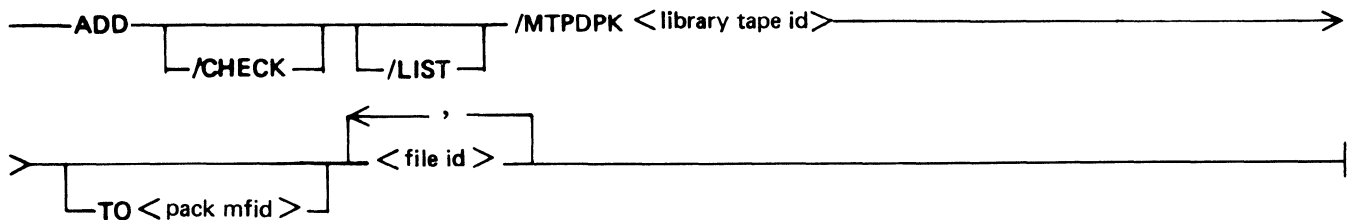


Figure B-6. ADD Command Syntax (DISK Tape to PACK)

ADD Option 6:

Option 6 invokes PACKUP to transfer files from a pack library tape to disk.



Figure B-7. ADD Command Syntax (Pack Tape to DISK)

CHECK

The CHECK command verifies the validity of a library tape. CHECK initiates LOADMP if files are to be checked on disk; PACKUP, if files are to be checked on pack. The syntax for CHECK is shown in Figure B-8.

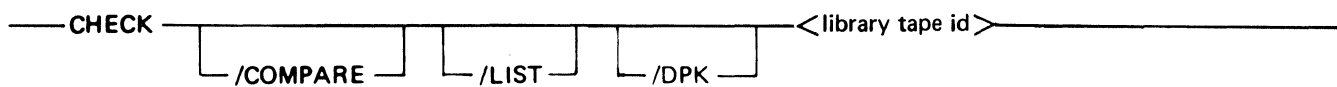


Figure B-8. CHECK Command Syntax

DUMP

The DUMP command copies disk or pack files to a library tape but does not remove the files. DUMP initiates LOADMP if the files are to be copied from disk; PACKUP if files are to be copied to pack. Figure B-9 shows the DUMP command syntax for initiating LOADMP to copy files from disk to library tape.

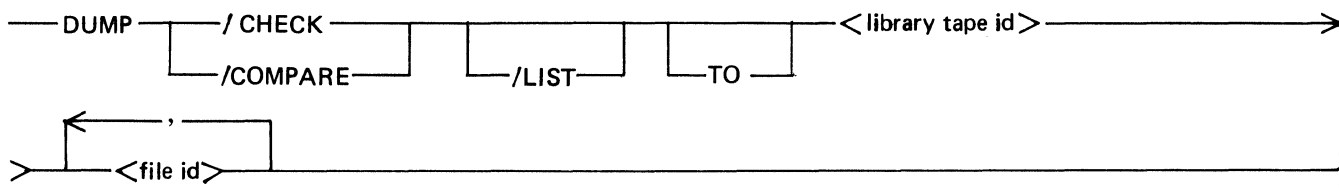


Figure B-9. DUMP Command Syntax (Disk)

Pack users can invoke PACKUP to DUMP files between disk, pack, and magnetic tape. Syntax diagrams for the four DUMP options appear in Figures B-10 through B-13.

DUMP Option 1:

Option 1 DUMPs files from disk to pack.

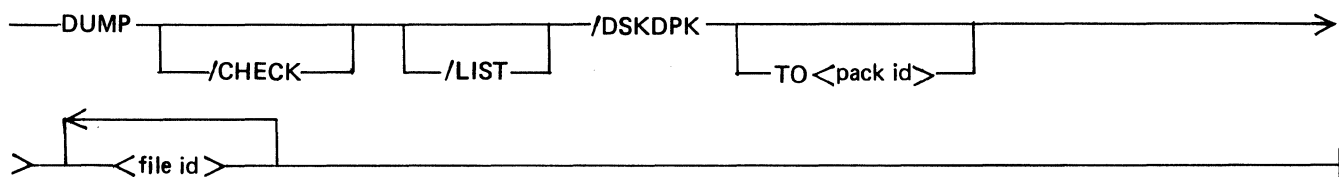


Figure B-10. DUMP Command Syntax (Disk to Pack)

DUMP Option 2:

Option 2 DUMPs files from pack to magnetic tape.

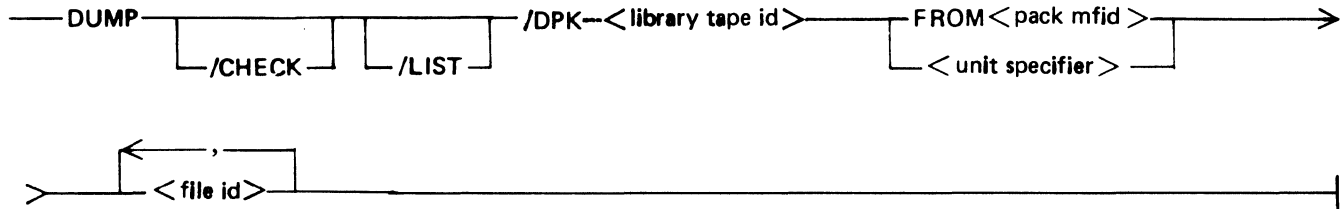


Figure B-11. DUMP Command Syntax (Pack to Tape)

DUMP Option 3:

Option 3 DUMPs files from pack to disk.

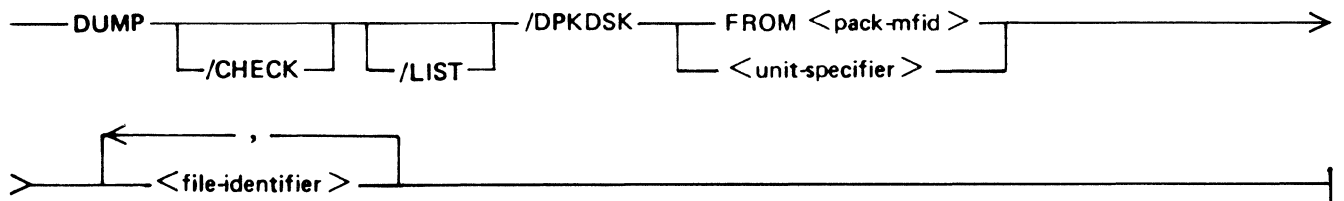


Figure B-12. DUMP Command Syntax (Pack to Disk)

DUMP Option 4:

Option 4 DUMPs files from pack to another pack.

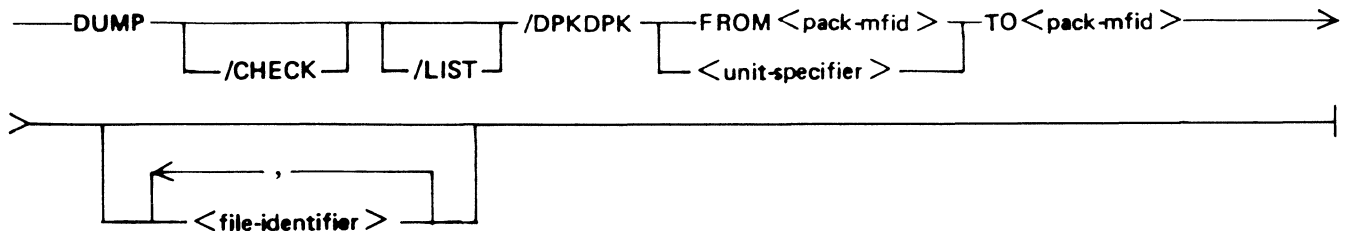


Figure B-13. DUMP Command Syntax (Pack to Pack)

LOAD

The LOAD command copies files from a library tape to disk or pack, replacing any existing file with any file on the tape having the same <file id>.

Figure B-14 shows the LOAD syntax that invokes LOADMP to copy files from a library tape to disk.

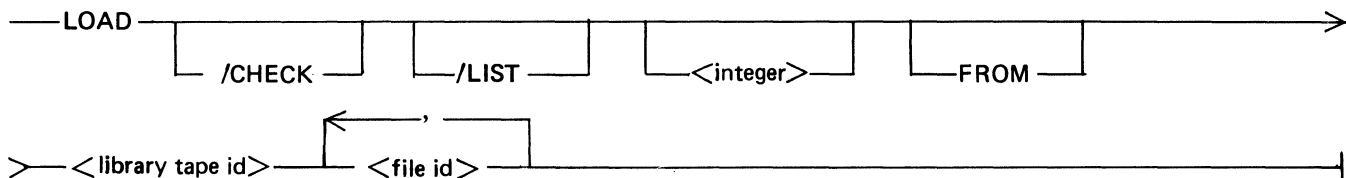


Figure B-14. LOAD Command Syntax (Disk)

LOAD Options for Packs

Additional facilities are provided for pack users to invoke PACKUP to LOAD files between disk, pack, and magnetic tape. The syntax for the LOAD instruction for packs is shown in Figures B-15 through B-20.

Option 1:

Option 1 transfers files from a pack library tape to a pack.

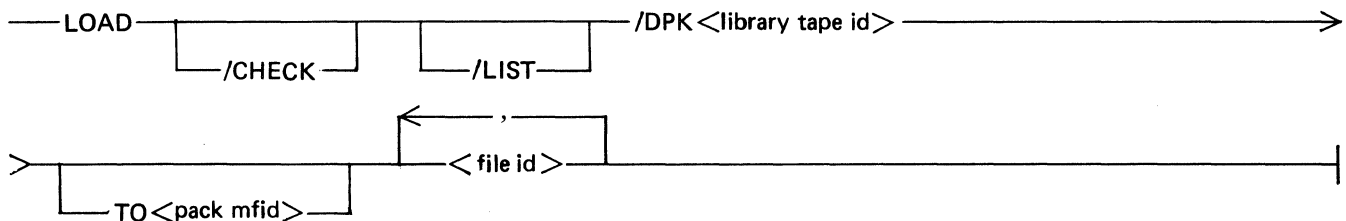


Figure B-15. LOAD Command Syntax (Tape to Pack)

Option 2:

Option 2 transfers files from pack to disk.

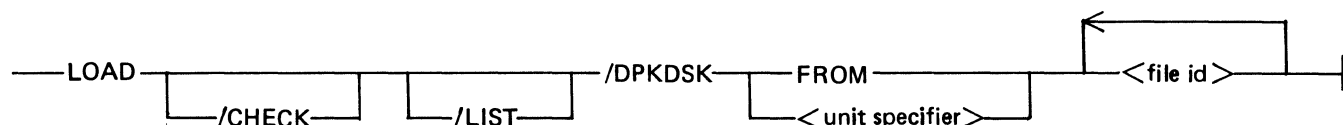


Figure B-16. LOAD Command Syntax (Pack to Disk)

Option 3:

Option 3 transfers files from disk to pack.



Figure B-17. LOAD Command Syntax (Disk to Pack)

Option 4:

Option 4 transfers files from pack to pack.

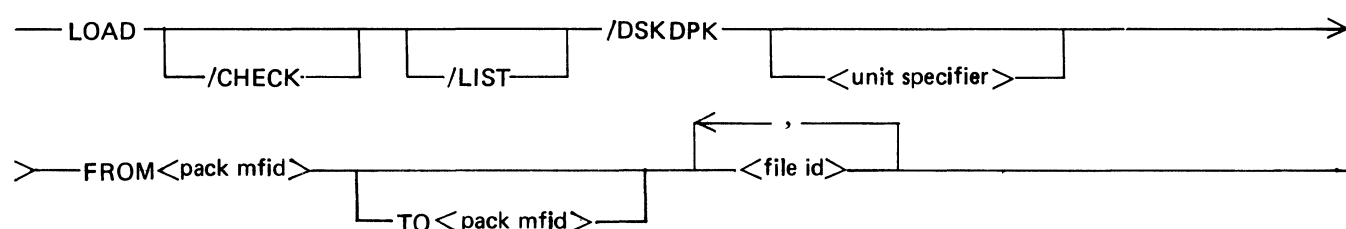


Figure B-18. LOAD Command Syntax (Pack to Pack)

Option 5:

Option 5 transfers files from a disk library tape to a pack.

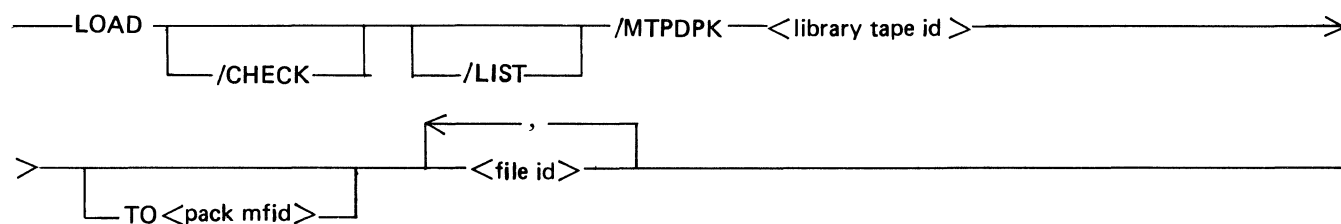


Figure B-19. LOAD Command Syntax (Tape to Pack)

The syntax shown in Figure B-20 files from a pack library tape to disk.

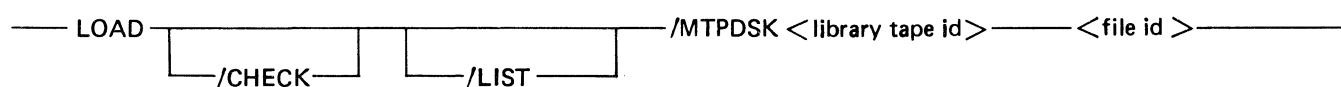


Figure B-20. LOAD Command Syntax (Tape to Disk)

UNLOAD

The UNLOAD command copies files from disk or pack to a library tape and removes the file-identifiers from the disk or pack directory after completing the operation.

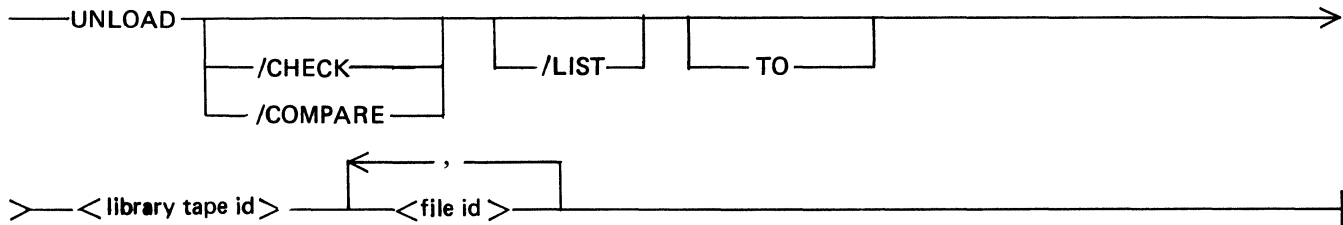


Figure B-21. UNLOAD Command Syntax

UNLOAD Option for Packs

Additional facilities are provided for pack users, to invoke PACKUP to UNLOAD files between disk, pack, and magnetic tape.

The syntax for the UNLOAD options for packs is shown in Figure B-22 through B-25.

Option 1:

Option 1 unloads files from pack to tape.

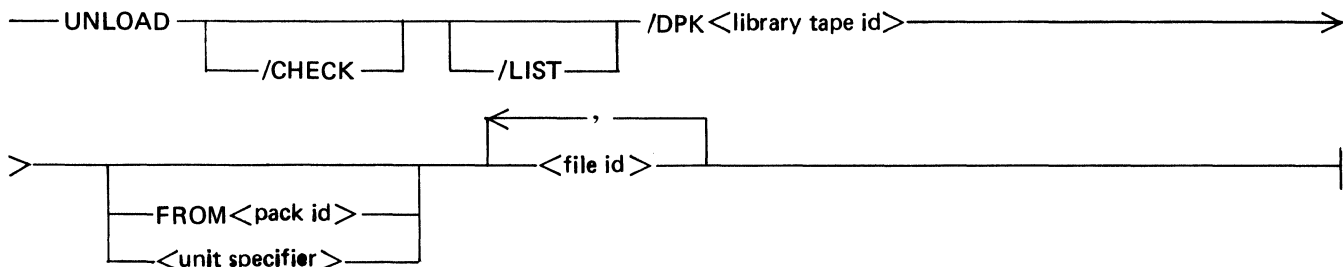


Figure B-22. UNLOAD Command Syntax (Pack to Tape)

Option 2:

Option 2 unloads files from disk to pack.

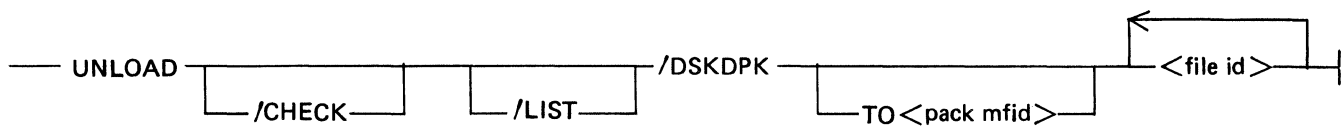


Figure B-23. UNLOAD Command Syntax (Disk to Pack)

Option 3:

Option 3 unloads files from pack to disk.

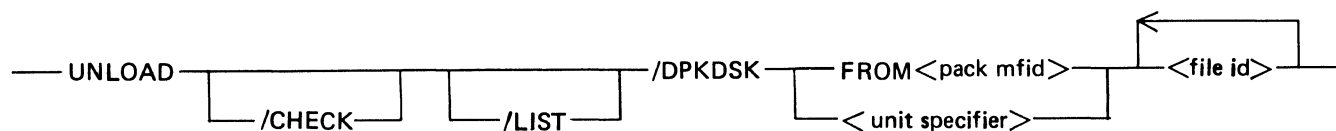


Figure B-24. UNLOAD Command Syntax (Pack to Disk)

Option 4:

Option 4 unloads files from pack to pack.

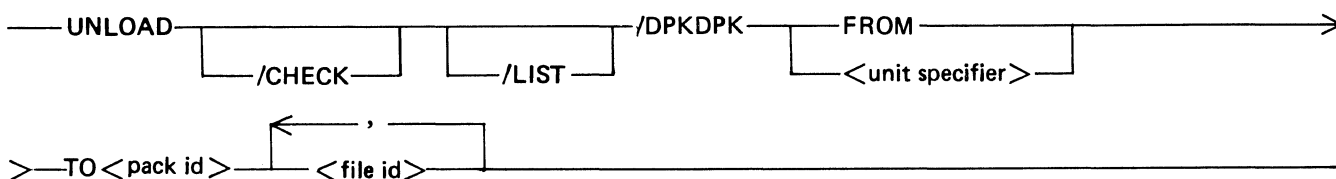


Figure B-25. UNLOAD Command Syntax (Pack to Pack)

PBDOUT And PBTOUT

The PB (Print Backup File) command initiates printing of printer backup files. If the printer backup file resides on tape, the program PBTOUT is scheduled. If the file resides on disk or pack, PBDOUT is scheduled.

The syntax of the PB command is shown in Figure B-26.

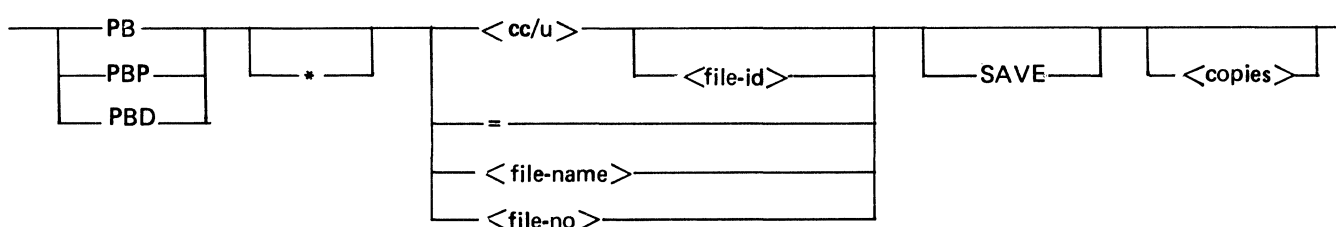


Figure B-26. PB Command Syntax



GLOSSARY OF TERMS

#. As the first character in a file name, identifies a pseudo card deck (control deck).

\$. As the first character in a file name, identifies a dump file.

*. As the first character in a file name, identifies a punch backup file.

@. As the first character in a file name, identifies a printer backup file.

=. *See* masking character.

5N Disk. A head-per-track storage medium which has a capacity of 5.5 MB.

ACCEPT statement. A statement in a program that causes the program to halt and wait for the appropriate data to be entered on the Operator Display Terminal (ODT). ACCEPT statements are available for BPL, COBOL ANSI-74, COBOL ANSI-68, FORTRAN, Pascal, and Work Flow Language (WFL) programs. *See also* AX (Respond to an ACCEPT) command.

access code. An identification code consisting of a user code, a password, and a charge number that a user enters when logging on to the system, or when entering a job initiation command. Normally used only on systems that have System Security (refer to *V Series System Software Security Installation and Operations Guide*). *See also* user code, password, charge number.

access level. A security level established for each remote terminal and/or access code. In the Command and Edit (CANDE) timesharing system, the access level for each station is selected in the USERHO deck. Each system command can be entered only from a terminal with a particular access level (or with a higher access level).

acronym. A word made up of the first letters (or some of the first letters) of a group of words. For example, ACU is the acronym for automatic calling unit.

active mix. Those jobs in the mix that are not waiting.

ACU. *See* Automatic Calling Unit.

adapter ID. A name used to identify a data communications line connected to the system with a Uniline DLP. *Contrast with* station name.

after-linkage. A condition where a specified task is scheduled to start only after another specified task has been completed. Refer to the AFTER (Hold Program) command.

alphanumeric. Any character in the system's character set.

American Standard Code for Information Interchange (ASCII). A coded character set of 128 control and graphic characters in which each character is represented by a seven-bit code (eight bits including parity check). ASCII is defined by American National Standard X3.4-1967.

area. The smallest amount of disk space occupied by a file. The size of an area is user-defined.

ASCII. *See* American Standard Code for Information Interchange.

asynchronous. Transmission at an arbitrary rate. A character, or block of characters, is surrounded by start and stop bits from which a receiver derives the necessary timing. *Contrast with* synchronous.

asynchronous process. A procedure initiated by a program that runs independently of and in parallel with the initiating program.

auditing. In DMSII, keeping track of each transaction in order to ease recovery from a failure.

AUHL option. *See* automatic halt/load option.

automatic calling unit (ACU). A device that permits a modem or terminal to dial calls automatically, and thereby establish a dialed link over a communications network.

automatic halt/load (AUHL) option. An option that causes an automatic halt/load when a system failure occurs, unless the failure occurs during a halt/load.

available table. A table maintained by the MCP that keeps track of the space available on each disk or pack in the system.

B 20. A generic term for the B 21 - B 27 family of Unisys workstations.

B 874 Data Communications Processor (DCP). A Data Communications Processor (DCP) that uses code and tables stored in its memory to manage and service terminals on up to 32 half-duplex lines.

B 9399 dual string controller. A dual string controller that supports up to eight B 9494-12 (or B 9494-24) disk spindles. Each of the disk spindles contains two logical volumes, so that a single B 9399 dual string controller can support up to 16 volumes.

B 974 Data Communications Processor. A Data Communications Processor (DCP) that provides up to three times the line connectivity of the B 874 Data Communications Processor.

backup. (1) A copy of a file that can be used if the original becomes corrupted. (2) A description of a file temporarily stored on disk or pack which is output to a unit-record device (printer, card punch). (3) To create a backup file.

bad spots. Unusable sectors on a disk or pack.

base area. The first area of a pack file. The base areas for all files of a pack family are contained on the master pack.

base clear. An electronic signal generated by the switch on the base of a V Series universal console used to initialize a Data Link Processor (DLP) and associated hardware. *Contrast with* master clear.

base pack. *See* master pack.

base relative address. An address within a program that is an offset from the beginning of the program rather than an absolute memory address.

BCL. A Unisys common character set in which each character is represented by a six-bit code.

BCT. *See* branch communicate.

BDLC Data Link Control. A bit-oriented data link control protocol that is compatible with the High-Level Data Link Control (HDLC) protocol defined by the International Standards Organization (ISO). BDLC stations support two-way simultaneous (TWS) operation over dedicated, full-duplex lines, and two-way alternate (TWA) operation over switched, half-duplex lines.

Beginning of Job (BOJ). The start of job processing.

Beginning of Tape (BOT). A reflective strip of tape placed on an edge of the backing side of magnetic tape that indicates the beginning of the usable portion of the tape. The reflective tape is placed 15-25 feet from the physical beginning of the tape. *Contrast with* end of tape (EOT).

binary configuration. Binary addressing of sectors on disk or pack.

Bisync. Unisys Binary Synchronous Communication protocol, a character-oriented data link control protocol similar to the IBM BSC protocol. Bisync is designed for two-way alternate (TWA) operation over either point-to-point or multipoint links. *Contrast with* BDLC.

bisync procedures. Binary synchronous procedures. Specific procedures that use synchronous transmission.

bit. Binary digit. The most basic unit of computer information. A bit can have one of two values: 0 (off or false) or 1 (on or true).

bits per inch (BPI). (1) The information-packing density of magnetic tape. (2) The density at which a magnetic tape unit can read and write.

blackout feature. A security feature that prevents unauthorized persons from viewing a particular access code as it is entered during the log-in procedure. *See also* the BO (Blackout Access Code) command.

block. (1) Physically-adjacent records that are transferred to or from a device as a group. (2) In data communications, a group of characters or bits sent as a unit.

blocking factor. The number of records per block.

Block Lockout Table (BLT). A table that keeps track of which file blocks are locked by which programs. The BLT resides in memory and is updated by the Master Control Program (MCP) every time a program locks or unlocks a file block. The BLT LIMIT record sets the maximum number of entries this table can have.

blocksize. A file attribute that gives the length of a block.

BLT. *See* Block Lockout Table.

BLT contention. A condition that occurs when one program wants to access a file block that is locked by another program running on the same processor. *See also* contention.

BNA Architecture. An overall set of design principles (that is, an architecture) under which Unisys systems can be joined into data communications networks. Each Unisys system that can be incorporated into a network has specific hardware and/or software conforming to this architecture.

BOJ. *See* Beginning of Job.

BPI. *See* bits per inch.

BPL Programming Language. An ALGOL-like language for V Series.

Branch Communicate (BCT). A V Series machine-instruction that causes control to be transferred from a user program to the MCP.

Branch Reinstall Virtual (BRV). An instruction used to reinstall a user task. It is executed only by the MCP's dispatching routine in the MCP kernel.

BRV. *See* Branch Reinstall Virtual.

buffer. An area in which data is stored temporarily for transfer from one storage medium to another.

byte. (1) Eight consecutive bits. (2) In X.25 terminology, an octet. *See also* bit, kilobyte, octet.

CANDE. *See* Command and Edit system.

CANDE Editor files. Files of the Editor program, which is a part of the Command and Edit (CANDE) system.

card. (1) A punched card used with a card reader to input the system configuration files. (2) *Synonym for* record when describing configuration records and control commands. *See also* card deck, card reader, system initialization.

card deck. More than one punched card; this term usually refers to a number of cards comprising the data input to a program.

cardless system. A system that uses flexible disk operations instead of card reader operations for system initialization.

card reader. An input device that reads data from punched cards. *See also* card, card deck, system initialization.

CCITT. *See* International Telegraph and Telephone Consultative Committee.

cc/uu. Represents the channel and unit address of a peripheral device. The cc is a one- or two-digit channel designation, and the uu is a one- or two-digit unit designation. The slash is required; spaces are not permitted. Tape drive unit numbers generally begin with 1; all other device designations usually begin with 0.

Central Processor Unit (CPU). The hardware unit that executes instructions.

channel. (1) In data communications, a communication path used to transmit signals between two or more points. (2) The Data Link Processor (DLP) in the processor cabinet or stand-alone I/O cabinet that handles communication along a path. *See also* Data Link Processor.

chargeable time. An amount calculated from direct time, input/output (I/O) time, and the current job's share of system overhead.

charge number. Charges for system use are assigned to this number. The charge number is assigned by the system administrator. *See also* access code. Refer to the *V Series System Software Security Installation and Operations Guide*.

clobber. A condition when one file supplants another. The new file is said to clobber the old file.

code file. A file produced by a compiler when a program is compiled. The file contains instructions in machine code. A code file is also referred to as an object code file.

cold start. The process by which the Master Control Program (MCP) is loaded to system disk. A cold start reinitializes all data structures on a disk, causing any information about existing files on that disk to be lost. *See also* initialization.

Command and Edit (CANDE) system. CANDE is a timesharing environment that provides generalized file preparation and updating capabilities and task control in an interactive, terminal-oriented environment. Refer to the *V Series CANDE Installation and Operations Guide*.

compile-and-go operation. A compile operation where no permanent code file is produced.

concurrent file access. The ability of two or more programs to simultaneously read and update the same file.

CONFIG#. The name of the system configuration file. The symbol # represents the system number.

CONFIG. *See* Configuration File Utility (CONFIG).

Configuration File Utility (CONFIG). The interactive utility that allows a user to create and modify a system configuration file, which is used to cold start a system. The Configuration File Utility can accept input from the following media: flexible disk, terminal, card reader, disk file, or pack file. Refer to the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

console DLP. A Data Link Processor (DLP) used by the system to access the ODT, flexible disk drive, DLP test bus, and remote links.

contention. (1) A situation that occurs when multiple tasks compete for exclusive use of a resource, such as an input/output (I/O) device or a record in a file. (2) In data communications, a situation where multiple users vie for the right to use a transmission channel. (3) A situation that occurs when two or more tasks attempt to lock the same block in a file.

contention mode. In data communications, a line condition in which no station is designated as a master station, and each station on the line must monitor signals on the line and wait for an idle condition before bidding for master status.

- contention overtime** The period of time a program can contend for a file block. See also stalemate.
- contiguous** Continuous, with no space in between.
- contiguous items** In COBOL, items that are described by consecutive entries in the DATA DIVISION and that have a definite hierarchical relationship to each other.
- continuation area** An area in a pack file that is allocated on a pack other than the pack on which the base area was originally located.
- control command** (1) A Command And Edit (CANDE) input message that begins with a specific character, usually a question mark (?), and is used to control or interrogate the CANDE operating environment. A control command can be entered from any attached CANDE station. (2) An MCP control instruction.
- control deck** A disk or pack file that contains 96 bytes per record, and six or eight records per block; a control deck generally contains the information needed to compile or execute a program. Synonym for pseudo card deck.
- control instruction** See MCP control instruction.
- controller** Provides the interface between the Data Link Processor (DLP) residing in the host and the mass storage units (such as disk, pack, and tape units).
- control point** In DMSII, a point in time at which no user program is in a transaction state and all buffers are written to pack.
- core-to-core transfer (CRCR)** A method of transferring information between executing programs. Each programming language has its own syntax for the CRCR feature. CRCR transfer is synchronous; one program offers information in a core-to-core transfer and waits until another program retrieves it. The first program cannot continue processing until the information is retrieved. Refer to the V Series Program Interfaces Programming Reference Manual.
- CP 2000 communication processor** A communication processor (CP) that connects stations to a host system. The CP 2000 can be used to serve terminals for single or multiple host systems.
- CP 368X communication processor** A communication processor (CP) that can provide data communications and message control services to several processors concurrently. A CP 368X is declared as a Front End Processor (FEP).
- CP 9580 Data Communications Processor** A Data Communications Processor (DCP) that provides communications over a high-speed connection.
- CPU** See Central Processor Unit.
- CRCR** See core-to-core transfer.
- CTLDCK** Control deck. A reserved file identifier. CTLDCK is the name of the physical card deck used by the LDCNTL utility to build a pseudo card deck. Refer to the V Series MCP/VS System Software Operations Guide Volume 3: System Utilities
- cylinder** All the tracks on all the platters of a disk or pack that have the same radius.
- DASDL** See Data and Structure Definition Language.
- Data and Structure Definition Language (DASDL)** A language used by a Database Administrator to describe the physical and logical characteristics of a data base, and the criteria to be used to ensure the integrity and security of the data contained within it.
- database** An integrated, centralized system of data files that several applications can access and update concurrently. See also Database Management System II.

Database Management System II (DMSII) DMSII is a network type database system with hierarchical features, and with features that can mimic attributes of a relational database system. Refer to Related Product Information at the beginning of this manual for a list of DMSII-related documentation.

Database Program (DBP) A part of the Database Management System (DMSII) that performs all functions necessary for accessing and updating the database to which it is attached. See also production DBP.

Data Circuit-terminating Equipment (DCE) A device connected to Data Terminal Equipment (DTE) on one end and to an X.25 network on the other end. The DCE serves as the entry point to the X.25 network, similar to a modem.

data comm Synonym for data communications.

data communications The bidirectional transfer of data between one or more devices and a central computer.

data communications device (1) A device that routes information between the computer and peripheral devices, such as Data Communications Processors (DCPs), Data Link Processors (DLPs), and uniline controllers. (2) A terminal.

Data Communications Processor (DCP) An intelligent front-end processor (such as the B 874, B 974, or CP 3680) that controls a data communications network, reducing the workload of the Central Processor.

Data Link Processor (DLP) A processor that serves as the system interface to a peripheral device, a peripheral controller, or data communications lines.

Data Terminal Equipment (DTE) (1) In X.25, the functional unit of a data station that establishes, maintains, and releases a connection and provides code and signal conversion between the data station and the transmission line. (2) User equipment, such as a host computer, front-end processor, concentrator, cluster controller, intelligent terminal, or any network entry access point. (3) An end-user device that connects to the Data Circuit-terminating Equipment (DCE), which either generates or receives data carried by the network.

DBP See Database Program.

DCE See Data Circuit-terminating Equipment.

DCOM (1) The data communications system option. (2) The name of the module that carries out data communications functions.

DCP See Data Communications Processor.

DCP code file A file containing microcode loaded into a Data Communications Processor (DCP).

DDP See Device Dependent Port (DDP).

DEBUG Refers to the interactive DEBUG module used for debugging user programs or the operating system.

decimal configuration Refers to decimal sector-addressing on disk or pack. Contrast with binary configuration.

deck number (1) The last four digits of a pseudo card reader file identifier. (2) The full five-digit portion of the file identifier used to reference a deck created on a specified system of a multiple system configuration.

default The value automatically given to a variable when no other value has been assigned.

device A hardware device, such as a disk drive, that writes data to or reads data from a medium. See medium.

device dependent port (DDP) The physical connection of the controller to the drives and the host system.

- diagnostics** A test designed to detect malfunctions and specify their causes.
- digit** (1) An integer from 0 to 9, inclusive. (2) A grouping of four consecutive bits, representing decimal 0 to 9 and hexadecimal A to F.
- directory** A table of contents listing the files contained on a device. The device is usually a disk, a pack, or a tape.
- direct time** See processor time.
- disk** A mass storage unit consisting of 100 bytes per sector.
- diskette** See industry compatible minidisk.
- disk file** A file stored on 100-byte-per-sector disk media. The characters per record and records per block (blocking factor) can be set by the programmer.
- disk file name** See file identifier.
- diskpack** See pack.
- DLP** See Data Link Processor.
- DMPMEM** A program used to create a raw memory dump on magnetic tape. Support personnel use the dump to analyze system problems. See dump file.
- DMSII** See Database Management System II.
- drive** The hardware device on which a disk is mounted.
- DTE** See data terminal equipment.
- dual host** A feature available on a disk controller that will allow two different Data Link Processors (DLPs) to access that path (but not simultaneously).
- dual port** A device that links a disk or pack unit to two separate controllers, providing path redundancy.
- dump file** A file containing the contents of memory, often created at the time of a failure for purposes of analysis.
- EBCDIC** See Extended Binary Coded Decimal Interchange Code.
- ECM Log** A system log file maintained on Unisys V 500 systems. The ECM log contains information about temperature, voltage and other conditions on the system. The data in the ECM log originates in the Environmental Control Module (ECM) of the V 500. The ECM log is maintained on the hard disk of the system console and is transferred to the host system by system commands.
- End of File (EOF)** The physical limit of a data file.
- End of Job (EOJ)** (1) A message that a job has finished executing. (2) In X.25, a control code that tells the receiver that a job is completed.
- End of Tape (EOT)** A reflective strip of tape placed on the inside edge of the backing side of the magnetic tape that indicates to the system that the tape is near the end of the reel. The reflective tape is placed no less than 25 feet from the physical end of the tape.
- End-of-transmission character (ETX)** A character used to signal the end of input (in EBCDIC "03").
- EOF** See end of file.
- EOJ** See end of job.
- EOT** See end of tape.
- ETX character** See end-of-transmission character.
- exchange** A device that enables multiple DLPs to be routed to multiple units (usually disk, pack, or tape).

execution time The time required for a task to run to completion. Execution time is also referred to as run time, or in COBOL, as object time.

Extended Binary Coded Decimal Interchange Code (EBCDIC) A coded character set in which each character is represented by an 8-bit code.

extension module A part of the operating system that carries out a specialized function and that can be called into memory by the user when this function is needed. Refer to the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

family name A name that identifies a set of mass storage volumes. See volume. See also pack family.

fault tolerance The ability of a system to continue processing despite the loss of a key component of that system.

FEP See Front-end Processor.

FIB See File Information Block.

file attributes A set of variables that specify the characteristics of a file (for example, record size, blocking factor, media types).

file block contention A situation where one program attempts to lock a block that is already locked by another program. The former program is suspended and must wait until the block is unlocked. The suspended program is said to be contending.

file header A collection of file attributes on disk or pack that contains the physical characteristics of a file, such as the name of the file and the number of records per block.

file identifier The name contained in the directory for a device, such as a tape, disk, or pack. A file identifier is six characters long (with trailing blanks provided), but cannot be made up of all nulls, all blanks, or all zeros.

File Information Block (FIB) A block of data generated by a compiler that describes a file to the MCP. When a program is compiled, each source language file declaration causes the compiler to generate a block of data about the file. This area, known as the File Information Block, contains flags, counters, addresses, and sizes used by the MCP during all phases of file processing. The MCP modifies the File Information Block during program execution.

file name (1) In RPG, a name or word that designates a set of data items. (2) In COBOL, a user-defined word that names a file described in a file description entry or in a sort-merge file description entry within the FILE SECTION of the DATA DIVISION. (3) Often used as synonym for file identifier. See also file identifier.

file number (1) The last four digits of the identifier of a printer, punch backup file or memory dump. (2) The full five-digit portion of the file identifier used to reference a file created on a system in a multi-system shared configuration.

firmware Microcode stored in a hardware device and used to control its operation.

FLAME Functional Logical Analysis of Machine Efficiency. A program that reports on V Series performance.

forms mode A mode of operation for a terminal in which the screen is divided into data fields (where user input is allowed) and display fields (where the user is unable to input).

frame (1) In bit-oriented protocols, the sequence of contiguous bits bracketed by and including opening and closing flag sequences (01111110). (2) A group of bits sent over a communications channel; generally a logical transmission unit sent between data-link-layer entities that contains its own control information for addressing and error checking.

Front-end Processor (FEP) Refers to a device that controls peripherals or that handles data communications. See also controller, Data Communications Processor, Data Link Processor.

full-duplex Two-way, simultaneous transmission.

GCR Density See Group Coded Recording (GCR) Density.

Generalized Message Control System (GEMCOS) A Message Control System that can run on Unisys V Series systems to provide generalized file preparation and updating capabilities and task control from an ODT. Refer to the B 2000/B 3000/B 4000/V Series GEMCOS Implementation and Programming Reference Manual.

GEMCOS See Generalized Message Control System.

Gregorian date The most common method of expressing the date (month/day/year or day/month/year). Contrast with Julian date.

Group Coded Recording (GCR) Density 6250 bytes per inch. GCR tapes can be written and read only by a tape drive that is GCR-capable.

group id A file identifier that can have terminating or embedded equal signs as masking characters to indicate a group of files. See also masking character.

guard file A disk or pack file that contains information on the security access rights of another disk or pack file.

half-duplex Two-way transmission that occurs in only one direction at a time.

halt/load (H/L) A system-initialization procedure that temporarily halts the system and loads the Master Control Program (MCP) from disk or pack to main memory. A halt/load is used to start the machine and to recover from failures. During the halt/load process, the tables and lists used by the operating system are built in memory. Refer to the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

Halt/load Supervisory Printer Operator Display Terminal (HLSPO) An Operator Display Terminal (ODT) that displays halt/load messages and fail codes in the event of system failure. The HLSPO can be declared in the UNIT ODT record of the system configuration file, and serves as the terminal for system initialization and the Fault Handler.

HC2-DLP See Inter-system Control.

header link The pointer to a file's header from its directory entry.

hit rate The percentage of times that a user program overlay was called and found in the user QWIK pool (that is, did not have to be retrieved from disk or pack).

H/L See halt/load.

HLSPO See Halt/load Supervisory Printer Operator Display Terminal.

host The central mainframe computer in a data communications network. The host is the computer system containing the Message Control System (MCS) to which a given station is connected.

host attributes Those attributes that are not common across a network but are applied on a host-by-host basis.

host components Those components that are not common across a network but are applied on a host-by-host basis.

hostname The logical name of a system. The hostname is used to identify the processor in the Run Log, the ODT Log, the Maintenance Log, and in a BNA network.

Host Transfer Data Link Processor (HT-DLP) A Data Link Processor that connects the host to the controller for disk and pack drives.

HT-DLP See Host Transfer Data Link Processor.

hung DLP A Data Link Processor (DLP) that detects an irrecoverable error during an input/output (I/O) operation. The DLP is stopped (hung) and must be cleared.

I/O See input/output.

ICM Independently Compiled Module. A component of a computer program compiled independently of the main program.

ICMD See industry compatible minidisk.

ID (1) An identifier, usually a name or number, of a file or storage device. (2) The unit designation that specifies the logical number of the pack or disk device. See also identifier.

ICPV A hardware mnemonic that represents a CP 2000 Communication Processor. ICPV is used in several commands, configuration records, and utility programs to declare options for the CP 2000.

identifier In COBOL, a data-name followed by the syntactically correct combination of qualifiers, subscripts, and indexes necessary to make unique reference to a data item.

ILPS See Intelligent Laser Printing System.

industry compatible minidisk (ICMD) A flexible disk that conforms to industry standards.

initialization See system initialization.

Initialize, Verify, and Relocate (IVR) A complete pack initialization process in which the pack controller performs the following: writes sector addresses and gaps in all sectors on the pack (initialization); reads and checks all of the addressable sectors on the pack for address errors and protection code errors (verification); for any errors found, writes the error sector's address into a spare sector address field, and writes the predefined data pattern into the sector's data area (relocation).

input/output (I/O) An operation in which the system reads data from or writes data to a file on a peripheral device such as a disk.

input/output base module A frame or tray that houses Data Link Processors (DLPs). The input/output base module contains the logic used to provide priority-resolution among requesting DLPs.

input/output (I/O) path (1) The code in the MCP executed when an I/O is requested. (2) The route input/output (I/O) takes to or from a peripheral device through a channel, a Data Link Processor (DLP), or any controllers or exchanges that lead to and from the peripheral.

Input/Output Processor (IOP) An asynchronous processor that controls the data flow between main memory and peripherals.

input/output subsystem (1) The portion of the Master Control Program (MCP) and other system facilities that are primarily concerned with physical and logical input/output (I/O) operations. (2) The hardware and software that manages all transfers of information between the MCP and peripheral devices.

input/output wait time The accumulated time spent waiting for input/output (I/O) operations to finish.

Intelligent Laser Printing System (ILPS) A software system that produces printouts on a laser image printer.

interface. (1) A communication path between two units. (2) A set of conventions for passing data and/or information. (3) In Pascal, the name for a list of program elements that a module proposes to share with other modules or programs. Each interface is defined in the module in which its elements are declared and invoked in the program unit that will share the elements.

interlaced file storage. A disk or pack format in which the physical sectors on the storage medium are arranged in non-sequential order such that more than one or more rotations of the medium is necessary to read or write all sectors on a given track. *Contrast with* sequential file storage.

International Telegraph and Telephone Consultative Committee (CCITT). A committee of the International Telecommunications Union (ITU) that has generated such international standards as Recommendation X.25 for packet-switching networks and Recommendation X.3 for packet assembler-disassemblers.

Inter-system Control (ISC). A system that provides local communication between Unisys processors. The ISC provides a high-speed data transfer rate between systems through a peripheral Data Link Processor (DLP) connection.

intrinsic. A utility program bound to the Master Control Program (MCP), usually initiated by the MCP, system commands, or user programs. In many cases, the user can provide a program of the same name that is invoked instead of the MCP-intrinsic program.

IOP. *See* Input/Output Processor.

ISC. *See* Inter-System Control.

IVR. *See* Initialize, Verify, and Relocate Pack.

job. A program initiated by an EXECUTE, COMPILE, or other initiation command. A job can be in the job mix or the schedule, and can be *executing*, *waiting*, or *stopped*. Each job has its own task number. *Synonymous with* task.

job control command. System commands such as FILE (Equate File-Name With File-Identifier Or Medium), MEMORY (Allocate Memory For Program Execution), PRIORITY (Set Priority) that specify the execution attribute of a job.

job initiation command. A system command that starts the execution of a job or causes a job to enter the schedule for execution, such as COMPILE (Compile Program), DEBUG (Attach Program To DEBUG Session), or EXECUTE (Execute Library Program).

job mix. *Synonymous with* mix.

job number. *See* task number.

job schedule. *See* schedule.

job status. The status of a job in the mix. The status of a job at a given time could be one of the following: EXECUTING, WAITING, TERMINATING, SCHEDULED.

job transfer. A BNA function that allows a job to be transferred across a network, from one system to another.

Julian date. The method for expressing the date in which the days in a year are numbered consecutively. For example, January 1 is Julian date 001, January 2 is Julian date 002, and so forth. *Contrast with* Gregorian date.

kb. *See* kilobyte.

kd. *See* kilodigit.

kernel. The dispatcher and interrupt-handler of the MCP.

keyboard input message. One of the two types of MCP system commands that allow an operator to interface with the MCP. The other type of command is called an MCP control instruction. Keyboard input messages have a wide range of functions, such as inquiry and system maintenance. *See also* MCP control instruction.

keyword. A reserved word that must be present when the syntax in which the word appears is used.

kilobyte. One thousand bytes (2000 digits).

kilodigit. One thousand digits (500 bytes).

label. The beginning or ending record of a mass storage volume or unit record device file.

LCN. Logical channel number.

LDCNTL. A system utility that builds pseudo card decks from physical card decks. Refer to the *V Series MCP/VS System Software Operations Guide Volume 3: System Utilities*.

LEM. *See* Line Expansion Module.

library tape. A tape created by the SYSTEM/COPY program with format and labels suitable for storing files.

Line Expansion Module (LEM). A cabinet that houses line adapters that cannot fit inside the B 974 Data Communications Processor cabinet.

line printer. In general, a printer that prints a line at a time. *See also* train printer.

LINC. Logic Information Network Compiler. A fourth generation compiler for a language which defines complete data processing systems.

link. A physical connection between two hosts, or between a host and a communications device.

LOCK. (1) A shared file I/O operation that locks a block, preventing other programs that are using shared I/O operations from updating it. The block remains locked until it is unlocked. (2) A program can be locked by the LOCK (Lock Program) command or the LP (Lock Program) command when the program is executed. A locked program cannot be discontinued. This prevents unauthorized or inadvertent interference. A program is unlocked by the UP (Unlock Program) command. (3) In DMSII, the LOCK or MODIFY statement in the host language reads a record from a data set and locks the record to prevent concurrent modification by another user.

log. A record of system activity.

log transfer. The process of closing one log file and opening a new log file.

logical channel. *See* virtual circuit.

logical channel address. In data communications, either the logical channel number (LCN), the data terminal equipment (DTE) address, or the Network Definition Language (NDL) station used to reference the logical channel.

logical record size. The amount of data accessed in the execution of one read or write statement in a program.

logical subsystem. A group of one or more 100-byte disk devices from one or more physical subsystems. Refer to the DISK record in the System Initialization section of the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

Magnetic Ink Character Recognition (MICR). A check-encoding system that automates check handling. Checks are imprinted with magnetic-ink characters of a typeface and dimensions specified by the American Banking Association. Fourteen different characters (the numbers 0-9 and four special symbols) provide amount, identification, and control information.

mailbox An inter-system, message-passing mechanism used to synchronize systems for MCP functions requiring exclusive shared resource access. The mailbox resides in the shared system processor (SSP). There can be only one mailbox per shared configuration.

maintenance log A file maintained by the Master Control Program (MCP) that contains a record of hardware and activity failures. Refer to the V Series System Software Logging Operations Reference Manual.

maintenance panel A "soft" panel consisting of screens displayed on the operator display terminal (ODT) used by support to load, display, and otherwise manipulate machine state, hardware registers, memory, index registers, and so forth.

Maintenance Processor A device that provides operator and maintenance interfaces to a V Series system. The Maintenance Processor allows local or remote field engineering personnel to run various levels of diagnostic programs.

masking character (=) An equal sign can be used as a masking character. The equal sign stands for any character. For example, the specification FIL=2 is relevant to all of the following: FIL02, FIL12, FILA2, FILB2.

MAST See Memory Area Status Table.

Master Available Table A special disk or pack file created at initialization time that contains a list of all usable space on the disk or pack.

master clear An electronic signal used to initialize a Data Link Processor (DLP) and associated hardware. Contrast with base clear.

Master Control Program (MCP) The V Series operating system. The MCP is the heart of the V Series system. It is designed to make optimum use of all system resources to maximize productivity and processing efficiency.

master pack The pack which contains the base areas for all files of a pack family.

MB See megabyte.

MC7 A Unisys program product, the B 774 Message Control System Generator, in which a network definition prepared by the user is the basis for generating a message control system that handles data communication functions.

MCP See Master Control Program.

MCP control instruction Along with keyboard input messages, one of the two types of MCP system commands that allows you to interface with the MCP. The MCP control instructions generally affect only a program in the mix, in that they either initiate or modify a job.

MCP QWIK memory Memory reserved for Master Control Program (MCP) overlays. The most recently used MCP overlays are kept in QWIK memory so that they do not have to be accessed from disk when they are needed. Refer to the LIMIT QWKMEM record in the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

MCP tables A set of arrays, maintained by the Master Control Program (MCP), which assists in controlling job execution and allocating system resources.

MCS See Message Control System.

medium (pl. media) A piece of hardware, such as a disk spindle, capable of storing data. A device writes data to or reads data from a medium. See device.

Medium Systems Network Definition Language (MSNDL) A Unisys program that generates Message Control System (MCS) or Data Communication Handler (DCH) programs that run on the host and handle data communication functions.

megabyte (MB) One million bytes.

Memory Area Status Table Data maintained by the manager of the Master Control Program and to record the state of allocated memory.

memory dump A copy of the contents of memory, sometimes referred to as a memory image. The system produces a memory dump when a problem occurs, so that the problem can be analyzed.

menu A formatted screen that presents the user with a set of selections.

Message Control System (MCS) An environmental, software program that is executed by the Central Processor (the host). The MCS controls the local data communications network consisting of the host, the terminals, and possibly a network controller such as a DCP. The purpose of the MCS is to protect user programs from having to perform data communication tasks. The MCS routes each message between a specific terminal and a specific user program. More than one MCS can be on a host, performing different functions. GEMCOS is an example of a V Series Message Control Systems.

MICR See Magnetic Ink Character Recognition (MICR).

microcode See firmware.

microminidisk drive (MMDD) A drive that is used to read and write to 3.5 inch disks.

minidisk See industry compatible minidisk.

minidisk image file A disk file containing the exact data as resident on the flexible disk.

mix The jobs or tasks that are currently executing. A list maintained by the Master Control Program (MCP) of all jobs that have begun execution. Contrast with schedule.

MIXED An option on UNIT NRZ and UNIT MPE indicating that the two unit declarations actually refer to the same physical drive, accessed in a mutually exclusive manner.

MIX table A table used by the MCP to maintain information about all programs in the mix or schedule. The type of information includes, wait indicators (that is, why a program cannot continue), various time calculations (such as stopped and average RUN/WAIT time and time waiting I/O) and program status.

mix time The length of time a job is in the mix.

MMDD See microminidisk drive.

mnemonic An easy-to-remember abbreviation, usually of a computer command or type of device.

MP See maintenance processor.

MPE See phase-encoded magnetic tape.

MT9 A nine-track ASCII-encoded magnetic tape. See Non-return to Zero.

MTC A hardware mnemonic representing magnetic cartridge tape devices. Currently these devices support a density of 38,000 bits per inch.

multifile identifier Archaic term for familyname.

NCP See Network Communications Port.

Network Communications Port A logical entry point into a host system. NCPs are part of the V Series Communication System (VCS).

Network Log A system log file maintained on V Series systems that use the V Series Communication System (VCS). The network log (sometimes referred to as the "NLOG") contains information about errors and security violations that occur within the VCS data communications network.

Non-return to Zero (NRZ) 800 bytes per inch on magnetic tape.

Non-status Device (NST) A device or unit that can be handled only by specialized programs, not by the Master Control Program (MCP).

Not Ready A hardware state recognized by the Master Control Program (MCP). A peripheral device such as a line printer or a magnetic tape drive can be in Ready, Not Ready, or Saved state.

NRZ See Non-return to zero.

NST See Non-Status Device.

numeric The digits 0 through 9.

object code file See code file.

OCS See Operator Control Station.

octet Eight contiguous bits (one byte). An octet is also referred to as a character.

ODT See Operator Display Terminal.

ODT log A file maintained by the Master Control Program (MCP) that contains all the messages that are displayed on the Operator Display Terminal. Synonymous with SPO log, SLOG. Refer to the V Series System Software Logging Operations Reference Manual.

Operator Control Station (OCS) This type of terminal is supported by a Uniline Data Link Processor. Contrast with Operator Display Terminal.

Operator Display Terminal (ODT) The video terminal through which the operator enters system commands and receives system messages. This type of terminal is supported by the Console Data Link Processor. Contrast with Operator Control Station.

overlay Logical divisions of a program that occupy the same memory area at different periods during the execution.

pack A mass storage unit consisting of 180 bytes per sector. Contrast with disk.

pack configuration An arrangement of pack units. A naming convention is used when discussing packs. Such terms as 2x8 (two-by-eight) or 4x16 (four-by-sixteen) are used in reference to controllers and exchanges and the number of pack units on a subsystem. The first number refers to the number of I/O paths (Data Link Processors) coming from the host system to the controller. The second number refers to the number of spindles (not drives) linked to that controller. The maximum number of spindles on a single subsystem is 16. The maximum number of paths to a single subsystem is dependent on the model.

pack controller A stand-alone unit that controls data flow between the Data Link Processor (DLP) in the host system and the individual pack spindles.

pack directory See directory table.

pack drive A cabinet containing one or two pack spindles. See also pack spindle.

packet In data communications, a unit of data routed from one Packet Switched Exchange (PSE) to another by transmission line. A packet consists of data surrounded by packet-level headers and trailers. A message can be broken down into packets at one side of the network and reassembled into the original message at the other end.

Packet Assembler-Disassembler (PAD) Software that allows a teletype-like terminal to send and receive data on a packet-switching network such as an X.25 network.

Packet Switching Exchange (PSE) In data communications, the switching nodes in a Public Data Network.

Packet Switching Network (PSN) A data communication network that transmits data by dividing messages into smaller units (packets), transmitting the packets separately to their destinations, and reassembling the message at the receiving end. A PSN is also referred to as a Public Switched Network.

pack family Two or more packs that have the same name.

pack file A file stored on a pack.

pack firmware Special firmware required by the pack controller.

pack initialization A process that makes packs usable. Packs are initialized by the DISPKV or PTD utility programs.

pack label Fields written at a special location on the pack at initialization used to identify the pack. Label fields include the serial number, system access code, pack name, system interchange code, and owner's ID among others.

pack spindle The hardware device on which a pack is mounted.

pack unit A pack spindle.

PAD See Packet Assembler-Disassembler.

page overflow A condition that occurs if the amount of information to be displayed exceeds the room available on a screen. The message "More Data" indicates that the extra information can be displayed by pressing the space bar and the transmit key. Page overflow cannot occur for an MSG table display.

parallel transmission A technique used to send an entire character at one time by using separate channels or separate frequencies on one channel. Contrast with serial transmission.

parameter A quantity or item that can be given a different value each time a process is repeated.

parameter list A collection of one or more parameters.

password Normally used only on systems that have System Security. A group of characters by which the system determines the legality of a user when a user logs on to the system or enters a job initiation command. The password is assigned by the system administrator. Refer to the V Series System Security Installation and Operations Guide.

patch file A file containing the replacement, insertion, or deletion records that are used to update an existing source file.

path See input/output path.

PDN (Public Data Network) A packet-switched network defined as a common carrier by government regulation. See Public Data Network.

PE See phase encoded.

peripheral A device external to the CPU and I/O cabinet. Common peripherals are magnetic tape drives, packs, and printers.

peripheral dependent firmware Special code loaded into the Universal Console or Maintenance Processor (MP) that allows the Processor to use the Industry Compatible Mini Disk (ICMD) drive or microminidisk drive (MMDD) as a peripheral.

Peripheral Switch II Provides the ability to switch peripherals between two, three, or four V Series systems. Peripheral switching is controlled either by a hand-held unit or by a terminal command.

phase encoded (PE) A type of data transmission where the phase of the carrier shifts when the data switches from 0 (zero) to 1 or 1 to 0 (zero). See phase-encoded magnetic tape.

phase-encoded magnetic tape (MPE) A method of encoding information on a tape at 1600 bytes per inch (bpi). Some tape drives can read and write tapes only in phase-encoded mode.

V Series MCP/VS System Software Operations Guide, Volume 2

Glossary of Terms

physical subsystem. The disk drives, pack drives, or combination of disk and pack drives that are connected to the system through the same Data Link Processors (DLPs). This arrangement is also referred to as a *disk subsystem* or a *pack subsystem*.

physical tape number (PTN). A serial number to identify a particular magnetic tape. The PTN is part of the tape label. Sometimes referred to as the *can number*.

pool size. The amount of memory available for user QWIK memory.

port. A point of data entry into a computer, network, or other electronic device.

printer backup file. A printer file stored for later use. Output is sent to disk, pack, or tape when a printer is not available, to increase efficiency, or when on-line viewing is desired.

printer files. When a program makes a listing, such as a report, the text of the listing and the codes that control the printer are a printer file. A printer file stored for later use on disk, pack, or tape is known as a *printer backup file*.

priority. (1) A characteristic associated with a job that determines its precedence in the use of system resources. (2) In X.25, the sequence in which various entries and tasks are processed by the analysis of action codes and other priority real-time systems.

privileged mode. A program execution mode in which the processor allows the execution of privileged instructions. Usually reserved for the operating system.

processor. The unit that executes instructions of the Master Control Program (MCP) and user programs.

processor name. A name given to a processor by the HOSTNAME record in the system configuration file.

processor number. A number given to each system. This number is used to make certain file names (for example, printer backup files) unique in a multi-system environment.

processor time. The accumulated time a processor has spent working on a job. *Synonym for direct processor time, direct time.*

Production DBP. A database program that binds a COBOL program into the DBP, allowing the DBP to access internal file information and record areas without involving the MCP. Other host language programs can access the database through the Production DBP.

program. A series of instructions for the computer to carry out.

programmatic. Relating to, or contained in, a program.

programmatic switch. A feature in some languages, such as COBOL, that allows commands to be given to a program during execution.

program name. The name of the object code file. A program name can have up to six characters; no punctuation is allowed. The first character must be a letter of the alphabet.

PSE. *See* Packet-Switching Exchange.

PSN. *See* Packet Switching Exchange.

pseudo card deck. *See* control deck.

Public Data Network (PDN). The most recently used Master Control Program (MCP) overlays are kept in QWIK memory so that they do not have to be accessed from disk when they are needed. Refer to the LIMIT QWKMEM record in the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

punch backup file. A file containing the same information as a punch file, but held on disk or pack for punching when desired. *See* punch file.

punch file. A file of the type required for output on a card punch.

queue. A list of tasks awaiting processing.

QWIK disk. A feature that allows part of system memory to store files, greatly speeding up access time.

QWIK memory. Memory that is reserved for rapid access to overlays.

QWIK pool. Memory available for user QWIK memory.

railroad syntax diagram. A graphic representation of the syntax of a command, utility, or program.

RAM. *See* random access memory.

random access memory (RAM). A type of memory that allows the reading and writing of memory without regard to the location of the preceding read or write operation.

R/D. *See* result descriptor.

Ready. A state recognized by the Master Control Program (MCP). A peripheral device such as a line printer or a magnetic tape drive can be in Ready, Non-ready, or Saved state.

record. (1) A group of logically-related items within a file that are treated as a unit. (2) The minimum amount of data read from or written to a file on one execution of a read or write statement in a program.

release. (1) A method of closing a file. (2) A term used to indicate that a file is no longer being used by the program. (3) A particular version of a software package.

release level. A number assigned to a particular version of the Master Control Program (MCP), such as MCP/VS 2.0.

Remote Job Entry (RJE) System. A software package that allows jobs, data, and control commands to be sent from a remote terminal to a central system, and that allows output of data from the central system to be sent to remote peripherals. Refer to the *B 2000/B 3000/B 4000/V Series Remote Job Entry (RJE) Operations Guide*.

remote support. Diagnostic routines and dump analysis can be performed either locally or through modem by a remote support center.

reset. (1) To turn an option off. (2) To make the value of a bit zero.

restricted master pack. A master (base) pack that contains only files having a multi-file-id identical to the pack's label name.

restricted pack. A pack restricted in access to programs having a multi-file ID identical to the pack's label name.

result descriptor (R/D). A coded description of the results of an input/output (I/O) operation.

RJE. *See* Remote Job Entry System.

RJE handler. A program running on the Central Processor that carries out Remote Job Entry (RJE) functions.

rollout. An action that occurs if insufficient memory is available for a particular job. In this case, the job's codefile can be *rolled out* to disk. The codefile is stored on disk and the space it was using in memory is relinquished. Jobs with lower memory priorities are rolled out first. When memory becomes available again, the codefile is rolled back into memory.

RPG Report Program Generator.

run log A file maintained by the Master Control Program (MCP) that contains a record of all job and file activity on the system. Refer to the V Series System Software Logging Operations Reference Manual.

Saved A software state recognized by the Master Control Program (MCP). A peripheral device can be in Ready, Non-ready, or Saved state.

schedule A list kept by the Master Control Program (MCP) of jobs that have not yet started executing.

SCSI See Small Computer Systems Interface.

sector A sector is the minimum addressable area on a pack or disk. A disk sector is 100 bytes; a pack sector is 180 bytes. Sectors are also referred to as segments.

sector address The logical address of a sector on a given disk or pack.

segment See sector.

SENSITIVEDATA A file security attribute that indicates that a file contains confidential or sensitive data. When the file is removed, its data will be overwritten with random data.

sequential organization The logical file structure in which a record is identified by a predecessor-to-successor relationship established when the record is placed in the file.

serial transmission A technique of sending data one bit at a time, one after the other. Contrast with parallel transmission.

set (1) To turn an option on. (2) To make the value of a bit one. A logically-related grouping.

shared file I/O operations Concurrent file access is regulated by a special set of shared file I/O operations. By using these I/O operations, programs can lock, read, seek, and write file blocks in a way that prevents data corruption by other programs that are using the same I/O operations.

shared pack The PACK record in the system initialization file can declare a shared pack. A shared pack must be declared with the same logical ID number and the same shared system processor (SSP) name for each processor.

shared system - disk only Two to four V Series systems that share disk but do not share pack devices.

shared system - disk and pack Two to four V series systems linked together in such a way that they share all disk and one or more pack devices.

shared system initialization records Records to be included in the system initialization file for shared systems. These initialization records, such as the CONNECT, DLP, and SSP records, are used at cold-start time. They enable the Master Control Program (MCP) and other programs to perform concurrent file access and to share peripherals. Refer to the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

shared system - pack Two to four V series systems linked together in such a way that they can share packs but they do not share disk. Each system runs its own copy of the same Master Control Program (MCP).

Shared System Processor (SSP) A type of Data Link Processor (DLP) that links up to four V Series systems so that they can share files on common disk and pack units. The systems can run common software and use conventional file structures (with the exception of DMSII files).

Signed numeric A numeric field that contains an algebraic sign (or a default sign).

Small Computer Systems Interface A communications protocol used to communicate with peripheral devices, including magnetic tape devices and disk drives.

SN Signed numeric.

snap See snapshot.

snapshot The creation of a file that describes machine state at the time of a failure. Refer to the USE SNAP record in the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

soft device Any device that requires a special code (firmware) to be loaded before the device can become operational. See firmware.

SPO See Supervisory Printer Operator Display Terminal.

SPOM See SPOMESSAGE BCT.

SPOMESSAGE BCT A programmatic branch communicate that passes keyboard input messages to the MCP and requests that the response be returned to the requestor. Contrast with ZIP and ZIPSPO.

SSP See shared system processor.

SSP contention A condition that occurs when a program running on one V Series system wants to access a file block that is locked by a program running on another V Series system. (When the programs are running on the same system, this situation is called BLT contention.)

stacked deck A term for a condition that occurs when the system configuration files for a shared system are loaded from the same card reader, and the card deck that is put into the card reader contains the system configuration files for more than one system.

stalemate A situation in which two or more programs cannot continue processing because they have locked and are contending for mutually exclusive records.

start/stop transmission See asynchronous.

station (1) The combination of functional units constituting the Data Terminal Equipment (DTE), Data Circuit-terminating Equipment (DCE), and their common interface. (2) In the Interactive Datacomm Configurator (IDC), a data structure that describes the attributes of a physical terminal. (3) The logical representation for definition (1).

statistics In DMSII, a feature that keeps track of database use.

status (1) In data communications, the ability of a station, at a given instant, to transmit or receive data. (2) A job can have the status of EXECUTING, WAITING, or TERMINATING. (3) A function of the MCP that handles various, housekeeping duties on a repeating, periodic basis.

start bit Indicates the beginning of a character, or block of characters, in an asynchronous transmission. See asynchronous.

stop bit Indicates the end of a character, or block of characters, in an asynchronous transmission. See asynchronous.

STOQ See storage queue.

STOQUE See storage queue.

storage queue A storage queue, abbreviated to STOQ or STOQUE, is a buffer in memory used to transfer information from one program to another. It provides asynchronous inter-program communication. This means that if a program sends a message to another program not ready to accept the message, the message waits in the storage queue until it is accepted. Contrast with core-to-core.

string A connected sequence or group of characters.

Supervisory Printer Operator Display Terminal (SPO) An obsolete term for Operator Display Terminal (ODT).

switch See programmatic switch.

synchronous. Transmission at a fixed rate, where transmitter and receiver are synchronized. Start and stop bits are not required. *Contrast with asynchronous.*

syncpoint. In Data Management System II (DMSII), a point in time when no program is updating a data base, and a syncpoint record is written to the audit trail.

system configuration file. A file that describes the logical structure of the system. Refer to the *V Series MCP/VS System Software Operations Guide Volume 1: Installation*.

system disk. The 100-byte media used to store the Master Control Program (MCP) code file.

system initialization. The act of bringing the Master Control Program (MCP) into memory. The two methods of initializing a system are cold-start and halt/load. The cold-start process copies the MCP to system disk; the halt/load process does not. The halt/load process assumes the MCP is already on system disk.

system security. System access control and disk file security functions. Refer to the *V Series MCP/VS System Software Operations Guide Volume 1: Installation* and the *V Series System Software Security Installation and Operations Guide*.

system status tables. Tables that contain information about the status of peripheral devices and jobs.

system tape. The release tape that contains the Master Control Program (MCP) and all necessary files, such as disk pack firmware and utility programs, for the execution of the MCP.

TABS II. *See* Time Analysis and Billing System II.

tailored I/O. I/O routings written for a specific device and type of I/O operation. This leads to greater overall system throughput. Tailored I/O paths are available for all commonly accessed peripheral devices.

tape mark. A tape mark is written after the beginning label, before the ending label, and after the last ending label of a physical reel of magnetic tape.

TELCOM DLP. A low-cost Data Communications Processor (DCP).

termination. The end of a job.

Test Bus Interface. A Maintenance Processor (MP) interface that communicates with Data Link Processors (DLPs) through the maintenance card and provides a kind of "back door" entry into hung DLPs.

test routines. There are two kinds of test routines available for V Series processors: confidence routines and diagnostic routines. Confidence routines allow for a certain level of testing of mainframe and peripherals, but primarily are used to verify that a module is performing correctly. Diagnostic routines can be used in a non-functioning environment to accurately and quickly diagnose the cause of a failure so that it can be quickly corrected.

Time Analysis and Billing System II (TABS II). A program that provides analysis and billing functions.

timesharing. A way of managing terminals so that each terminal user seems to have continuous access to the system. CANDE is a timesharing system.

TRACE. An MCP service that provides a listing of a program's execution path, usually at a machine-instruction level.

train printer. A type of line printer that employs a moveable, metal chain of characters. Train printers require translation table files. *See* translation table file.

Train Printer Data Link Processor (TP-DLP). A processor that serves as the controller of a train printer and provides the input/output (I/O) interface between the system and the printer.

TRAK. A real-time debugging tool that records various MCP data structures or events. TRAK entries are stored in a TRAK buffer in memory.

transfer rate. The speed at which information can be transferred to the system depends on the speed of the peripheral device.

translate table file. Software that indicates the position of characters on a particular train printer. *See* train printer.

translation table. Software that provides one-to-one character translation (for example, from EBCDIC to ASCII).

TWA. *See* two-way alternate.

two-way alternate. *Synonymous with* half-duplex.

two-way simultaneous. *Synonymous with* full-duplex.

TWS. *See* two-way simultaneous.

UA. Denotes unsigned alphanumeric data stored in bytes. *Contrast with* UN.

UN. Specifies unsigned numeric data, which is stored as digits and undigits. *Contrast with* UA. *See* undigit.

undigit. The undigits are A (hexadecimal 10) through F (hexadecimal 15).

Uniline Data Link Processor (DLP). A data communications device that provides support for Unisys standard communication disciplines and terminal products on a single line. The Uniline DLP is designed for installations that need limited data communications connectivity.

Uniline firmware. A special code file loaded into the Uniline Data Link Processor (DLP) so that the device can communicate with the operator display terminal (ODT) or with other data comm devices.

unit. A single peripheral device.

UNLOCK. A shared file input/output (I/O) operation that unlocks a locked block. If a block is not locked (by a program) but the program tries to unlock it, the program is terminated.

unrestricted pack. A pack that is available to all users (also called system resource pack). *Contrast with* restricted pack.

Unsigned numeric. Numeric data stored without algebraic signs.

user code. Generally used only on systems that have System Security. The user code is a number that identifies a user to the system when a user logs on to the system or enters a job initiation command. The user code is assigned by the system administrator. *See* access code. Refer to the *V Series System Software Security Installation and Operations Guide*.

USERHO. In System Security, the program that creates the system file containing usercodes and passwords and their privileges. Refer to the *V Series System Software Security Installation and Operations Guide*.

user mode. The operating mode of applications programs, in which the processor senses all interrupts and an Input/Output Processor (IOP) or Input/Output Translator (IOT) can set the interrupt toggle to interrupt executing programs.

user program. A job executing on the host that makes use of the services of the processor and of the MCP and other system software.

user QWIK memory The user QWIK pool is memory dedicated to storing the most recently used user program overlays, so that they will not have to be accessed from disk if they are needed again. See also MCP QWIK memory. Refer to the USE QWIK record in the V Series MCP/VS System Software Operations Guide Volume 1: Installation.

utility A standard program that helps computer operation, such as initializing flexible disks or converting files from one medium to another.

V Series Communication System (VCS) Data communication software that connects a V Series system with data communication terminals. VCS software resides and executes on a V Series system. See also Network Communications Port (NCP) and Network Log.

version date The date the Master Control Program (MCP) was compiled.

virtual circuit A path between two devices in a Packet Switching Network.

Virtual Static Imaging Device (VSID) A device that produces high-speed, high-quality printouts (for example, a laser printer).

volume A volume is a named medium, which the MCP can recognize regardless of which device the medium resides on.

VSID See Virtual Static Imaging Device.

waiting A job can be in one of several states -- executing, waiting, stopped, and so forth. For example, a job can be in the waiting state because of insufficient memory, disk, or pack space.

WFL See Work Flow Language.

Work Flow Language (WFL) The Unisys language used for constructing jobs that compile and run programs. WFL includes variables, expressions, and flow-of-control statements that offer the programmer a range of task control capabilities. Refer to the B 2000/B 3000/B 4000/V Series WFL Compiler Programming Reference Manual.

Working Available Table A file maintained by the MCP that records space available on disk or pack. There is one table for all disks. Each pack maintains its own table.

writeahead In DMSII, a feature that causes buffers to be written to pack ahead of the point where the expected write is necessary.

X.25 An international, standard packet-switching network interface, designed to allow different computers to communicate.

ZIP A facility that allows user programs to send system commands. The commands that are sent with the ZIP facility are handled as if they had been entered at the operator display terminal (ODT). The ZIP facility is invoked by constructs in the programming languages.

ZIPSP0 A facility that gives a user program the ability to pass control information to the MCP and to determine if any errors occurred during the processing. If there is an error, a response will be displayed and returned to the program.



INDEX

Special Characters

Glossary-1

\$Glossary-1

* Glossary-1

@ Glossary-1

= Glossary-1

5N Disk Glossary-1

A

AC (Purge And Place Tape Number In Label) 2-2 to 2-3

ACCEPT Glossary-1

access code 2-340 to 2-341, Glossary-1

access level Glossary-1

restrictions 1-12

acronym Glossary-1

active mix Glossary-1

ACU Glossary-1

AD (Specify ODT/OCS Screen Parameters) 2-4 to 2-8

AD specification

commands 2-4

table display options 2-7

adapter ID Glossary-1

ADD (Copy Files From Library Tape To Disk or Pack) 2-9,
B-1 to B-3

AFTER (Hold Program) 2-10

after-linkage 2-259, Glossary-1

AFTR option 2-295

AJ (Display Active Jobs) 2-12

ALLOCATE (Create Disk Space And Directory Entries) 2-13 to
2-15

alphanumeric Glossary-1

ALTER (Change System Parameters) 2-16 to 2-20

American Standard Code for Information Interchange
(ASCII) Glossary-1

APBD option 2-295

APCR option 2-295

APCR, SD (Set Deck Limit) 2-285

area Glossary-1

ASCII Glossary-1

asynchronous Glossary-1

asynchronous process Glossary-1

AT (Initiate BNA Job Transfer) 2-22 to 2-23

AT (Pass Message To Remote Host) 2-21

auditing Glossary-1

AUHL Glossary-2

AUHL option 2-295

AURD option 2-295

AUTO KX 2-169

automatic calling unit (ACU) Glossary-2

automatic halt/load (AUHL) option Glossary-2

available table Glossary-2

AX (Pre-Respond To An ACCEPT) 2-25 to 2-26

AX (Respond To An ACCEPT) 2-24

B

B 20 Glossary-2

B 874 Data Communications Processor (DCP) Glossary-2

B 9399 dual string controller Glossary-2

B 974 Data Communications Processor Glossary-2

backup Glossary-2

backup files

BF, BFP (List Backup Files on Disk or Pack) 2-30 to 2-31

BFRF (Search, List, Remove Backup Files) 2-23 to 2-37

CD, CDP (List Inactive Pseudo Decks) 2-47 to 2-48

CV, CVP (Convert Punch Backup File To Pseudo Card
Deck) 2-74 to 2-75

PB (Print Backup File) 2-220

PC, PCP (Punch Backup File) 2-221 to 2-223

PRINT (Invokes PBDPRN) 2-243 to 2-250

RF, RFP (Remove Backup File From Disk or Pack)

SK (Skip Backup Records) 2-291

bad spots Glossary-2

base area Glossary-2

base clear Glossary-2

base pack Glossary-2

base relative Glossary-2

BCL Glossary-2

BCL (Mark Beginning Of Card Data File) 2-27

BCT 2-37, Glossary-2

BDLC Data Link Control Glossary-2

Beginning of Job (BOJ) Glossary-2

Beginning of Tape (BOT) Glossary-2

BEGINUSER (Mark Beginning Of Job Stream) 2-28 to 2-29

BF, BFP (List Backup Files on Disk or Pack) 2-30 to 2-31

BFRF (Search, List, Remove Backup Files) 2-32 to 2-37

BFRF intrinsic 2-37

binary configuration Glossary-2

BINARY (Read Binary Card Deck) 2-38

Bisync Glossary-2

bisync procedures Glossary-3

bit Glossary-3

bits per inch Glossary-3

blackout feature Glossary-3

block Glossary-2

Block Lockout Table, SHOW (Display System Information) 2-
286 to 2-289, Glossary-3

blocking factor Glossary-3

blocksize Glossary-3

BLT Glossary-3

BLT contention Glossary-3

BNA

architecture Glossary-3

AT (Initiate BNA Job Transfer) 2-22 to 2-23

V Series MCP/VS System Software Operations Guide, Volume 2

Index

BNA (Continued)

AT (Pass Message To Remote Host) 2-21
CA (List Port Level Candidates) 2-43 to 2-44
ENDAT (End Job Transfer) 2-122
HN (Host Name) 2-152 to 2-153
NET (Network Control Command) 2-194 to 2-196
NW (Networking Command Prefix) 2-199
SM (Send System Message) 2-292

BO (Blackout Access Code) 2-39

BOJ Glossary-3

BOJ option 2-295

BPI Glossary-3

BPL Programming Language Glossary-3

Branch Communicate(BCT) Glossary-3

Branch Reinstate Virtual (BRV) Glossary-3

BRV Glossary-3

BT (Begin TRAK) 2-40 to 2-41

buffer Glossary-3

BUFFER (Set MCS Result Queue Size) 2-42

buffered printer, adding a 2-333

byte Glossary-3

C

CA (List Port Level Candidates) 2-43 to 2-44

CANDE Glossary-3

entering commands through 1-7

QT (Quit Program Operation) 2-255 to 2-256

CANDE Editor files Glossary-3

card Glossary-3

BCL (Mark Beginning Of Card Data File) 2-25

BINARY (Read Binary Card Deck) 2-38

CV, CVP (Convert Punch Backup File To Pseudo Card Deck) 2-74 to 2-75

DATA/DATAB (Mark Beginning Of Card Data File) 2-78 to 2-79

EBCDIC (Mark Beginning Of Card Date File) 2-119

END/ENDCTL (End of Card File or Control Instructions) 2-121

PC, PCP (Punch Backup File) 2-221 to 2-223

SPO (Simulate SPO Through Card Reader) 2-300

card deck Glossary-3

cardless system Glossary-3

card punch, adding a 2-326

card reader Glossary-4

adding a 2-326

commands invalid from 1-12

cards, entering commands through 1-6

CC (Enter Control Instructions) 2-45 to 2-46

CCITT Glossary-4

cc/uu Glossary-4

CD,CDP (List Inactive Pseudo Decks) 2-47 to 2-48

Central Processor Unit (CPU) Glossary-4

CHANGE (Change File ID) 2-49 to 2-51

channel Glossary-4

ALTER (Change System Parameters) 2-16 to 2-20

DBIC (Initialize Controller Parameters) 2-84

XA (Add/Remove A Channel From An Exchange) 2-361

XC (Inhibit/Uninhibit Or Display Status Of A Channel) 2-362

CHARGE (Insert Charge Number) 2-52

chargeable time Glossary-4

charge number Glossary-4

CHECK (Verify Library Tape) 2-53, B-3

CHRG option 2-295

CK (Test Peripheral Device) 2-54

clobber Glossary-4

CLOS option 2-295

CN (Display Tape Number) 2-55

code file Glossary-4

cold start Glossary-4

Command and Edit (CANDE) system Glossary-4

command restrictions 1-12

COMPARE (Compare Files On Different Media) 2-56

compile-and-go operation Glossary-4

COMPILE (Compile Program) 2-57 to 2-62

concurrent file access Glossary-4

CONFG# Glossary-4

CONFIG Glossary-4

Configuration File Utility (CONFIG) Glossary-4

console DLP Glossary-4

contention Glossary-4

contention mode Glossary-5

contention overtime Glossary-5

contiguous items Glossary-5

continuation area Glossary-5

control command Glossary-5

control deck Glossary-5

control instruction Glossary-5

controller Glossary-5

control point Glossary-5

COPY (Invoke File Transfer Utility) 2-63 to 2-72

COPY option 2-295

COPYADD (Transfer Files) 2-73

core-to-core transfer (CRCR) Glossary-5

CP 3680 Data Communications Processor Glossary-5

CP 9580 Data Communications Processor Glossary-5

CPU Glossary-5

CRCR Glossary-5

CTLDCR 2-27, Glossary-5

CV, CVP (Convert Punch Backup File To Pseudo Card Deck) 2-74 to 2-75

cylinder Glossary-5

D

DA,DAP (Deactivate Pseudo Reader) 2-76 to 2-77

DASDL Glossary-5

DATA/DATAB (Mark Beginning Of Card Data File) 2-78 to 2-79

Data and Structure Definition Language Glossary-5

database Glossary-5

IR (Initiate DMSII Recovery) 2-161

V Series MCP/VS System Software Operations Guide, Volume 2

Index

database (Continued)

- SB (Interrogate Database Activity) 2-283 to 2-284
- SP (Display/Set/Change Database Parameters) 2-299

Database Management System II (DMSII) Glossary-5

Database Program (DBP) Glossary-6

Datacircuit-terminating Equipment (DCE) Glossary-6

data comm Glossary-6

- AT (Initiate BNA Job Transfer) 2-22 to 2-23
- AT (Pass Message To Remote Host) 2-21
- ENDAT (End Job Transfer) 2-122
- HN (Host Name) 2-152 to 2-153
- NET (Network Control Command) 2-194 to 2-196
- NW (Networking Command Prefix) 2-199
- SI (Set Remote Device Identification) 2-290
- SM (Send System Message) 2-292
- SS (Send Message) 2-305
- UNIT (Add Peripheral Devices) 2-325 to 2-335

data communications Glossary-6

data communications device Glossary-6

Data Communications Processor (DCP) Glossary-6

Data Link Processor (DLP) Glossary-6

Data Terminal Equipment (DTE) Glossary-6

DATE option 2-296

DBCA (Clear SSP Address Contention) 2-80 to 2-81

DBCL (Cancel DLP) 2-82

DBDR (Produce SSP Content Report) 2-83

DBIC (Initialize Controller Parameters) 2-84

DBMM (Set/Reset Maintenance Mode) 2-85

DBP Glossary-6

DBSR (Produce Block Lockout Table Report) 2-86

DBUA (Unlock Address) 2-87

DEBUG option 2-296

DBUL (Clear Locked Files on Shared Systems) 2-88

DC (Display Compilation Data) 2-89 to 2-90

DCE Glossary-6

DCOM Glossary-6

DCOM option 2-296

DCP Glossary-6

DCP code file Glossary-6

DCP option

adding a 2-327

BUFFER (Set MCS Result Queue Size) 2-42

LH (Load Firmware Into A Channel Controller or DCP) 2-176 to 2-178

SO (Set System Option) 2-296

UNIT (Add Peripheral Devices) 2-325 to 2-335

DD (Delete DLP) 2-91

DDP Glossary-6

DEBUG (Attach Program To Debug Session) 2-92 to 2-95, Glossary-6

debugging

BT (Begin TRAK) 2-40 to 2-41

CK (Test Peripheral Device) 2-54

DBCA (Clear SSP Address Contention) 2-80 to 2-81

DBCL (Cancel DLP) 2-82

DBDR (Produce SSP Content Report) 2-83

DBIC (Initialize Controller Parameters) 2-84

DBSR (Produce Block Lockout Table Report) 2-86

DBUA (Unlock Address) 2-87

DBUL (Clear Locked Files on Shared Systems) 2-88

DEBUG (Attach Program To Debug Session) 2-92 to 2-95

DISPLAY/WHATS (Display Information) 2-101 to 2-104

DM (Dump And Continue) 2-110 to 2-111

DP (Dump and Discontinue) 2-112

ED (Enter DEBUG Session) 2-120

ET (End TRAK) 2-126

GT (Initiate Trace) 2-148 to 2-149

ID (Invoke DEBUG Session) 2-154

MEMDUMP (Abnormal EOJ Memory Dump) 2-186

NT (Terminate Trace) 2-198

PM (Print Memory Dump) 2-230 to 2-238

QD (Query Debug) 2-254

TEST (Mark A Test Operation) 2-312

TRACE (Trace a Program) 2-321 to 2-322

decimal configuration Glossary-6

deck number Glossary-6

default Glossary-6

device Glossary-6

device dependent port (DDP) Glossary-6

diagnostics Glossary-7

digit Glossary-7

DIR (List File Directory) 2-99 to 2-100

directory Glossary-7

direct time Glossary-7

disk Glossary-7

ALLOCATE (Create Disk Space And Directory Entries) 2-13 to 2-15

DISK (Add A Disk) 2-96 to 2-98

DISPLAY/WHATS (Display Information) 2-101 to 2-104

RX (Return Deleted Space) 2-279 to 2-280

SQ (Squash Disk) 2-301

XD (Remove Disk Segments) 2-363

DISK (Add A Disk) 2-96 to 2-98

diskette Glossary-7

disk file Glossary-7

disk file name Glossary-7

disk maintenance

ADD (Copy Files From Library Tape To Disk or Pack) 2-9

KA (Analyze Disk Directory) 2-162 to 2-164

KP (Print Disk Segments) 2-165

KS (Analyze Disk Space) 2-166 to 2-167

KX (Obtain User Disk Space) 2-168 to 2-169

PATCH (Patch File) 2-215 to 2-219

REMOVE (Delete Files) 2-262 to 2-264

V Series MCP/VS System Software Operations Guide, Volume 2

Index

disk maintenance (Continued)

RF, RFP (Remove Backup File From Disk or Pack) 2-266 to 2-267

RM (Remove Duplicate File) 2-268 to 2-269

diskpack Glossary-7

disk/pack maintenance, CHANGE (Change File ID) 2-49 to 2-51

display information

AD (Specify ODT Screen Parameters) 2-4 to 2-7

AJ (Display Active Jobs) 2-12

BF, BFP (List Backup Files on Disk or Pack) 2-30 to 2-31

BFRF (Search, List, Remove Backup Files) 2-32 to 2-37

CA (List Port Level Candidates) 2-43 to 2-44

CD, CDP (List Inactive Pseudo Decks) 2-47 to 2-48

CK (Test Peripheral Device) 2-54

CN (Display Tape Number) 2-55

DBDR (Produce SSP Content Report) 2-83

DBSR (Produce Block Lockout Table Report) 2-86

DC (Display Compilation Data) 2-89 to 2-90

DIR (List File Directory) 2-99 to 2-100

DISPLAY/WHATS (Display Information) 2-101 to 2-104

DQ (Display System Messages) 2-113

DX (Display DLPs Attached To An Exchange) 2-118

ES (End Screen Display) 2-125

FI (List Pack Files In Use) 2-131

FN (List File Names) 2-142

HN (Host Name) 2-152 to 2-153

KA (Analyze Disk Directory) 2-162 to 2-164

KS (Analyze Disk Space) 2-166 to 2-167

KX (Obtain User Disk Space) 2-168 to 2-169

MX (Display Active Mix) 2-193

NET (Network Control Command) 2-194 to 2-196

OL (Display Peripheral Status) 2-203 to 2-208

OT (Display Program Data) 2-209

PA (Analyze Pack Directory) 2-212

PD (Print Directory) 2-224 to 2-225

PS (Analyze Pack Space) 2-253

QD (Query DEBUG) 2-254

SB (Interrogate Database Activity) 2-283 to 2-284

SHOW (Display System Information) 2-286 to 2-289

TI (Display Time Statistics) 2-313

TO (Type System Options) 2-319

WB (Display Bound Programs) 2-344

WC (Display Available Memory) 2-345

WD (Display Date) 2-346 to 2-347

WJ (Waiting Jobs) 2-348 to 2-349

WM (Display MCP Version) 2-350 to 2-351

WO (Display Active USER QWIK Operation Counts) 2-352 to 2-353

WQ (Display STOQUE Count) 2-354 to 2-355

WS (Display Jobs in Schedule) 2-356

WT (Display Time) 2-357

WX (Display Removed Space) 2-358

WY (Display Job Status) 2-359 to 2-360

XC (Inhibit/Uninhibit Or Display Status Of A Channel) 2-362

DISPLAY/WHATS (Display Information) 2-101 to 2-104

DL (Delete Peripheral Devices) 2-105

DLP Glossary-7

DBCL (Cancel DLP) 2-82

DBIC (Initialize Controller Parameters) 2-84

DD (Delete DLP) 2-91

DLP (Add DLPs) 2-106 to 2-109

DX (Display DLPs Attached To An Exchange) 2-118

DLP (Add DLPs) 2-106 to 2-109

DM (Dump And Continue) 2-110 to 2-111

DMPALL

GENERATE (Schedule DMPALL For Execution) 2-145 to 2-146

PERFORM (Schedule DMPALL) 2-226 to 2-227

QT (Quit Program Operation) 2-255 to 2-256

DMPANL 2-230 to 2-238, 2-255 to 2-256

DMPMEM Glossary-7

DMPOUT 2-230 to 2-238, 2-255 to 2-256

DMSII Glossary-7

DMS2 option 2-296

DP (Dump and Discontinue) 2-112

DQ (Display System Messages) 2-113

DR (Date Reset) 2-114

drive Glossary-7

DS (Discontinue Program) 2-115 to 2-116

DSKOUT 2-162 to 2-167, 2-255 to 2-256

DTE Glossary-7

dual host Glossary-7

dual port Glossary-7

DUMP (Copy Files To Library Tape) 2-117, B-3 to B-5

dump

DM (Dump And Continue) 2-110 to 2-111

DP (Dump and Discontinue) 2-112

DUMP (Copy Files To Library Tape) 2-117

MEMDUMP (Abnormal EOJ Memory Dump) 2-186

PM (Print Memory Dump) 2-230 to 2-238

dump file Glossary-7

DUP LIB 2-268 to 2-269

duplicate file 2-268 to 2-269

DX (Display DLPs Attached To An Exchange) 2-118

E

EBCDIC Glossary-7

EBCDIC (Mark Beginning Of Card Date File) 2-119

ED (Enter DEBUG Session) 2-120

END/ENDCTL (End of Card File or Control Instructions) 2-121

ENDAT (End Job Transfer) 2-122

End of File (EOF) Glossary-7

End of Job (EOJ) Glossary-7

End of Tape (EOT) Glossary-7

End-of-transmission character (ETX) Glossary-7

ENDUSER (Mark End Of Job Stream) 2-123 to 2-124

V Series MCP/VS System Software Operations Guide, Volume 2

Index

- EOF Glossary-7
- EOJ Glossary-7
- EOJ option 2-296, Glossary-7
- EOT Glossary-7
- ES (End Screen Display) 2-125
- ET (End TRAK) 2-126
- ETX character Glossary-7
- exchange Glossary-7
- EXECUTE (Execute Library Program) 2-127 to 2-130
- execution time Glossary-8
- Extended Binary Coded Decimal Interchange Code (EBCDIC) Glossary-8
- extension module Glossary-8
- F**
- family name Glossary-8
- fault tolerance Glossary-8
- FEP Glossary-8
- FI (List Pack Files In Use) 2-131
- FIB Glossary-8
- FILE (Equate File-Name With File-Identifier Or Medium) 2-132 to 2-140
- file attributes Glossary-8
- file block contention Glossary-8
- file handling
 - ALLOCATE (Create Disk Space And Directory Entries) 2-13 to 2-15
 - BFRF (Search, List, Remove Backup Files) 2-32 to 2-37
 - CHANGE (Change File ID) 2-49 to 2-51
 - DATA/DATAB (Mark Beginning Of Card Data File) 2-78 to 2-79
 - FILE (Equate File-Name With File-Identifier Or Medium) 2-132 to 2-140
 - FM (Direct File To Device — Special Forms Response) 2-141
 - FR (Designate Final Reel) 2-144
 - IL (Assign Labeled File) 2-155 to 2-156
 - MR (Remove Duplicate File) 2-191 to 2-192
 - OF (Optional File Indication) 2-200
 - OU (Direct File To Device) 2-210 to 2-211
 - PATCH (Patch File) 2-215 to 2-219
 - PERFORM (Schedule DMPALL) 2-226 to 2-227
- file header Glossary-8
- file-identifier 1-9 to 1-11, Glossary-8
 - reserved by MCP 1-9
- File Information Block (FIB) Glossary-8
- file name Glossary-8
- file number Glossary-8
- firmware Glossary-8
- FLAME Glossary-8
- FM (Direct File To Device — Special Forms Response) 2-141
- FN (List File Names) 2-142
- forms mode Glossary-8
- FP (Flush Program) 2-143
- FR (Designate Final Reel) 2-144
- frame Glossary-8
- Front-end Processor (FEP) Glossary-9
- full-duplex Glossary-9
- G**
- GCR Density Glossary-9
- GCR tape drive, adding a 2-330
- Generalized Message Control System (GEMCOS) Glossary-9
- GENERATE (Schedule DMPALL For Execution) 2-145 to 2-146
- GEMCOS Glossary-9
- GO (Restart A Stopped Program) 2-147
- Gregorian date Glossary-9
- Group Coded Recording (GCR) Density Glossary-9
- group id Glossary-9
- GT (Initiate Trace) 2-148 to 2-149
- GTDK option 2-296
- guard file Glossary-9
- H**
- half-duplex Glossary-9
- halt/load (H/L) Glossary-9
 - HL (Halt/Load Request) 2-150 to 2-151
- Halt/load Supervisory Printer Operator Display Terminal (HLSPO), Glossary-9
- HC2-DLP Glossary-9
- header link Glossary-9
- hit rate Glossary-9
- H/L Glossary-9
- HL (Halt/Load Request) 2-150 to 2-151
- HLSPO Glossary-9
- HN (Host Name) 2-152 to 2-153
- host Glossary-9
- host attributes Glossary-9
- host components Glossary-9
- hostname Glossary-10
- Host Transfer Data Link Processor (HT-DLP) Glossary-10
- HT-DLP Glossary-10
- hung DLP Glossary-10
- I**
- I/O Glossary-10
- I/O control
 - DBCL (Cancel DLP) 2-82
 - DBIC (Initialize Controller Parameters) 2-84
 - FP (Flush Program) 2-143
 - PRP (Change Processing And I/O Priority) 2-241 to 2-242
 - PRP (Set Processing And I/O Priority) 2-251 to 2-252
- ICM Glossary-10
- ICMD Glossary-10
- ID Glossary-10
- ID (Invoke DEBUG Session) 2-154
- identifier Glossary-10
- IL (Assign Labeled File) 2-155 to 2-156
- ILPS Glossary-10
- IN (Insert Data Into Program) 2-157 to 2-158
- industry compatible minidisk (ICMD) Glossary-10
- initialization Glossary-10

V Series MCP/VS System Software Operations Guide, Volume 2

Index

- Initialize, Verify, and Relocate (IVR) Glossary-10
- input/output (I/O) Glossary-10
- input/output base module Glossary-10
- input/output I/O path Glossary-10
- Input/Output Processor (IOP) Glossary-10
- input/output subsystem Glossary-10
- input/output wait time Glossary-10
- inquiry
 - AJ (Display Active Jobs) 2-12
 - AT (Initiate BNA Job Transfer) 2-22 to 2-23
 - AT (Pass Message To Remote Host) 2-21
 - AX (Pre-Respond To An ACCEPT) 2-25 to 2-26
 - AX (Respond To An ACCEPT) 2-24
 - BF, BFP (List Backup Files on Disk or Pack) 2-30 to 2-31
 - BFRF (Search, List, Remove Backup Files) 2-23 to 2-37
 - CA (List Port Level Candidates) 2-43 to 2-44
 - CD, CDP (List Inactive Pseudo Decks) 2-47 to 2-48
 - CN (Display Tape Number) 2-55
 - DC (Display Compilation Data) 2-89 to 2-90
 - DQ (Display System Messages) 2-113
 - FI (List Pack Files In Use) 2-131
 - HN (Host Name) 2-152 to 2-153
 - KA (Analyze Disk Directory) 2-162 to 2-164
 - KS (Analyze Disk Space) 2-166 to 2-167
 - KX (Obtain User Disk Space) 2-168 to 2-169
 - MX (Display Active Mix) 2-193
 - OL (Display Peripheral Status) 2-203 to 2-208
 - PD (Print Directory) 2-221 to 2-223
 - TI (Display Time Statistics) 2-313
 - TO (Type System Options) 2-319
 - WB (Display Bound Programs) 2-344
 - WC (Display Available Memory) 2-345
 - WD (Display Date) 2-346 to 2-347
 - WJ (Waiting Jobs) 2-348 to 2-349
 - WM (Display MCP Version) 2-350 to 2-351
 - WO (Display Active USER QWIK Operation Counts) 2-352 to 2-353
 - WQ (Display STOQUE Count) 2-354 to 2-355
 - WS (Display Jobs in Schedule) 2-356
 - WT (Display Time) 2-357
 - WY (Display Job Status) 2-359 to 2-360
- INSERT (Insert Data Into Program) 2-159 to 2-160
- Intelligent Laser Printing System (ILPS) Glossary-10
- interface Glossary-11
- interlaced file storage Glossary-11
- International Telegraph and Telephone Consultative Committee (CCITT) Glossary-11
- Inter-system Control (ISC) Glossary-11
- intrinsic Glossary-11
- intrinsic, WB (Display Bound Programs) 2-344
- introduction, system commands 1-1
- IOP Glossary-11
- IPP, adding an 2-328
- IR (Initiate DMSII Recovery) 2-161
- ISC Glossary-11
 - adding an 2-329
 - ISC option 2-296
- IVR Glossary-11
- J**
- job Glossary-11
- job control
 - AFTER (Hold Program) 2-10
 - GT (Initiate Trace) 2-148 to 2-149
 - LOCK (Lock Function) 2-184
 - LP (Lock Program) 2-185
 - MEMORY (Allocate Memory For Program Execution) 2-187 to 2-188
 - NL (Resume At New Location) 2-197
 - NT (Terminate Trace) 2-198
- job control command Glossary-11
- job initiation
 - COMPILE (Compile Program) 2-57 to 2-62
 - DEBUG (Attach Program To Debug Session) 2-92 to 2-95
 - DM (Dump And Continue) 2-110 to 2-111
 - EXECUTE (Execute Library Program) 2-127 to 2-130
 - GENERATE (Schedule DMPALL For Execution) 2-145 to 2-146
 - PB (Print Backup File) 2-220
 - PC, PCP (Punch Backup File) 2-221 to 2-223
 - PERFORM (Schedule DMPALL) 2-226 to 2-227
 - PM (Print Memory Dump) 2-230 to 2-238
 - PRINT (Invokes PBDPRN) 2-243 to 2-250
 - RA (Remove After-Linkage) 2-259
 - RERUN (Log a Rerun Operation) 2-265
 - RN, RNP (Activate Pseudo Reader) 2-270 to 2-271
 - RUN (Initiate Execution Of Program) 2-277
- job initiation command Glossary-11
- job maintenance
 - FILE (Equate File-Name With File-Identifier Or Medium) 2-132 to 2-140
 - FN (List File Names) 2-142
 - GO (Restart A Stopped Program) 2-147
 - IN (Insert Data Into Program) 2-157 to 2-158
 - INSERT (Insert Data Into Program) 2-159 to 2-160
 - PR (Change Priority) 2-241 to 2-242
 - PRIORITY (Common Priority Setting) 2-251 to 2-252
 - SHOW (Display System Information) 2-286 to 2-289
 - SK (Skip Backup Records) 2-291
 - ST (Suspend Program Processing) 2-306
 - START (Restart A Stopped Program) 2-307
 - STOP (Suspend Program Processing) 2-308
 - TRACE (Trace a Program) 2-321 to 2-322
- job mix Glossary-11
- job number Glossary-11
- job schedule Glossary-11
- job status Glossary-11
- job transfer Glossary-11

V Series MCP/VS System Software Operations Guide, Volume 2

Index

job termination

- DP (Dump and Discontinue) 2-112
- DS (Discontinue Program) 2-115 to 2-116
- FR (Designate Final Reel) 2-144
- MEMDUMP (Abnormal EOJ Memory Dump) 2-186
- QT (Quit Program Operation) 2-255 to 2-256
- RS (Remove Job From Schedule) 2-275
- TIME (Discontinue Program After a Specified Amount of Time) 2-314

Julian date Glossary-11

K

KA (Analyze Disk Directory) 2-162 to 2-164

kb Glossary-11

kd Glossary-11

kernel Glossary-11

keyboard input message Glossary-12

keyboard input messages vs. MCP control instructions 1-2

keyword Glossary-12

kilobyte Glossary-12

kilodigit Glossary-12

KP (Print Disk Segments) 2-165

KS (Analyze Disk Space) 2-166 to 2-167

KX (Obtain User Disk Space) 2-168 to 2-169

L

label Glossary-12

LABEL1 (Change Tape Label Format) 2-170 to 2-173

LC (Insert Log Comment) 2-174

LCN Glossary-12

LDCNTL 2-255 to 2-256

LD, LDP (Create Pseudo Card Deck) 2-175

LDCNTL Glossary-12

LD, LDP (Create Pseudo Card Deck) 2-175

LEM Glossary-12

LH (Load Firmware Into A Channel Controller or DCP) 2-176 to 2-178

LI (Log In) 2-179

LIB option 2-296

library maintenance

- ADD (Copy Files From Library Tape To Disk or Pack) 2-9
- CHECK (Verify Library Tape) 2-53
- COMPARE (Compare Files On Different Media) 2-56
- COPY (Invoke File Transfer Utility) 2-63 to 2-72
- COPYADD (Transfer Files) 2-73
- DIR (List File Directory) 2-99 to 2-100
- DUMP (Copy Files To Library Tape) 2-117
- EXECUTE (Execute Library Program) 2-127 to 2-130
- LOAD (Move Files From Library Tape to Disk or Pack) 2-183
- MERGE (Transfer Files From Multiple Sources, Resolving Duplicates) 2-189
- MOVE (Move Files To New Media) 2-190
- UNLOAD (Move Library Files Between Media, Remove Old File) 2-336

library tape Glossary-12

Line Expansion Module (LEM) Glossary-12

line printer Glossary-12

LINC Glossary-12

link Glossary-12

LN (Transfer and Print Log) 2-180 to 2-181

LO (Log Out) 2-182

LOAD (Move Files From Library Tape to Disk or Pack) 2-183, B-6 to B-7

LOADMP B-1

LOCK Glossary-12

LOCK (Lock Function) 2-184

log Glossary-12

log transfer Glossary-12

logging

CHARGE (Insert Charge Number) 2-52 to

LC (Insert Log Comment) 2-174

LN (Transfer and Print Log) 2-180 to 2-181

RERUN (Log a Rerun Operation) 2-265

TEST (Mark A Test Operation) 2-312

TL (Transfer Log) 2-315

logical channel Glossary-12

logical channel address Glossary-12

logical record size Glossary-12

logical subsystem Glossary-12

LP (Lock Program) 2-185

M

Magnetic Ink Character Recognition (MICR) Glossary-12

mailbox Glossary-13

maintenance log Glossary-13

maintenance panel Glossary-13

Maintenance Processor Glossary-13

manual

organization of xi

prerequisites xi

related product information xii

scope and purpose of xi

masking character (=) Glossary-13

MAST Glossary-13

Master Available Table Glossary-13

master clear Glossary-13

Master Control Program (MCP) Glossary-13

master pack Glossary-13

MB Glossary-13

MC7 Glossary-13

MCP Glossary-13

MCP control instruction Glossary-13

MCP control instructions vs. keyboard input messages 1-2

MCP QWIK memory Glossary-13

MCP memory, QWKMEM (Set MCP Overlay Buffer Pool) 2-258

MCP tables Glossary-13

MCS Glossary-13

medium (pl. media) Glossary-13

V Series MCP/VS System Software Operations Guide, Volume 2

Index

Medium Systems Network Definition Language
(MSNDL) Glossary-14

megabyte (MB) Glossary-14

MEMDUMP (Abnormal EOJ Memory Dump) 2-186

MEMORY (Allocate Memory For Program Execution) 2-187 to 2-188

memory

DM (Dump And Continue) 2-110 to 2-111

DP (Dump and Discontinue) 2-112

MEMDUMP (Abnormal EOJ Memory Dump) 2-186

MEMORY (Allocate Memory For Program Execution) 2-187 to 2-188

PM (Print Memory Dump) 2-230 to 2-238

PRM (Change Memory Priority) 2-241 to 2-242

PRM (Set Memory Priority) 2-251 to 2-252

QWKMEM (Set MCP Overlay Buffer Pool) 2-258

RQ (Remove Storage Queue Entries) 2-275

RX (Return Deleted Space) 2-279 to 2-280

WC (Display Available Memory) 2-345

XM (Remove Memory From System) 2-364

Memory Area Status Table Glossary-14

memory dump Glossary-14

menu Glossary-14

MERGE (Transfer Files From Multiple Sources, Resolving Duplicates) 2-189

Message Control System Glossary-14

MICR Glossary-14

MICR option 2-296

microcode Glossary-14

microminidisk drive (MMDD)

minidisk Glossary-14

minidisk image file Glossary-14

mix Glossary-14

AJ (Display Active Jobs) 2-12

ALTER (Change System Parameters) 2-16 to 2-20

DQ (Display System Messages) 2-113

MX (Display Active Mix) 2-193

RS (Remove Job From Schedule) 2-276

MIXED Glossary-14

MIX TABLE Glossary-14

MIX TIME Glossary-14

MMDD Glossary-14

mnemonic Glossary-14

MOVE (Move Files To New Media) 2-190

MP Glossary-14

MPE Glossary-14

MPE tape drive, adding an 2-330, Glossary-14

MR (Remove Duplicate File) 2-191 to 2-192

MT9 Glossary-14

MT9 tape drive, adding an 2-330, Glossary-14

multifile identifier Glossary-14

MX (Display Active Mix) 2-193

N

NET (Network Control Command) 2-194 to 2-196

NL (Resume At New Location) 2-197

Non-return to Zero (NRZ) Glossary-14

Non-status Device (NST) Glossary-14

Not Ready Glossary-15

NRZ Glossary-15

NST Glossary-15

NST, adding an 2-331

NT (Terminate Trace) 2-198

numeric Glossary-15

NW (Networking Command Prefix) 2-199

O

object code file Glossary-15

OCS Glossary-15

adding an 2-332

octet Glossary-15

ODT Glossary-15

AD (Specify ODT/OCS Screen Parameters)

adding an 2-332

CC (Enter Control Instructions) 2-45 to 2-46

entering commands through the 1-3

ES (End Screen Display) 2-125

LI (Log In) 2-179

LO (Log Out) 2-182

MCP response to ODT input 1-4

SS (Send Message) 2-305

ODT log Glossary-15

OF (Optional File Indication) 2-200

OK (Continue Processing Suspended Program) 2-201 to 2-202

OL (Display Peripheral Status) 2-203 to 2-208

OPEN option 2-296

Operator Control Station (OCS) Glossary-15

Operator Display Terminal (ODT) Glossary-15

OT (Display Program Data) 2-209

OU (Direct File To Device) 2-210 to 2-211

overlay Glossary-15

QWIKPOOL (Set Program Overlay Buffer Pool) 2-257

P

PA (Analyze Pack Directory) 2-212

PACK (Add A Pack) 2-213 to 2-214

pack Glossary-15

ALTER (Change System Parameters) 2-16 to 2-20

DBMM (Set/Reset Maintenance Mode) 2-85

PA (Analyze Pack Directory) 2-212

PACK (Add A Pack) 2-213 to 2-214

PO (Power Off Pack) 2-239

PP (Print Pack Sectors) 2-240

PS (Analyze Pack Space) 2-253

RX (Return Deleted Space) 2-279 to 2-280

SQP (Squash Pack) 2-302 to 2-304

XP (Remove Pack Sectors) 2-365

pack configuration Glossary-15

pack controller Glossary-15

pack directory Glossary-15

V Series MCP/VS System Software Operations Guide, Volume 2

Index

- pack drive Glossary-15
- packet Glossary-15
- Packet Assembler-Disassembler (PAD) Glossary-15
- Packet Switching Exchange (PSE) Glossary-15
- Packet Switching Network (PSN) Glossary-15
- pack family Glossary-16
- pack file Glossary-16
- pack firmware Glossary-16
- pack initialization Glossary-16
- pack label Glossary-16
- pack maintenance
 - REMOVE (Delete Files) 2-262 to 2-264
 - RF, RFP (Remove Backup File From Disk or Pack) 2-266 to 2-267
 - RM (Remove Duplicate File) 2-268 to 2-269
- pack spindle Glossary-16
- pack unit Glossary-16
- PACKUP B-1
- PAD Glossary-16
- page overflow Glossary-16
- parallel transmission Glossary-16
- parameter Glossary-16
- parameter list Glossary-16
- password Glossary-16
- PATCH (Patch File) 2-215 to 2-219
- patch file Glossary-16
- PATCHK 2-218 to 2-219
- path Glossary-16
- PB (Print Backup File) 2-220
- PBD option 2-296
- PBDOUT 2-220, B-1, B-9
- PBDPRN 2-243 to 2-250, 2-255 to 2-256, B-1
- PBP option 2-296
- PBPS option 2-296
- PBT option 2-296
- PBTOUT 2-220, B-1, B-9
- PC, PCP (Punch Backup File) 2-221 to 2-223
- PCD option 2-296
- PCH option 2-296
- PCHOUT 2-255 to 2-256
- PCP option 2-296
- PCRM option 2-296
- PD (Print Directory) 2-224 to 2-225
- PDN (Public Data Network) Glossary-16
- PE Glossary-16
- PERFORM (Schedule DMPALL) 2-226 to 2-227
- peripheral Glossary-16
- peripheral dependent firmware Glossary-16
- peripheral maintenance
 - AC (Purge and Place Tape Number In Label) 2-2 to 2-3
 - AD (Specify ODT/OCS Screen Parameters) 2-4 to 2-8
 - PG (Purge Magnetic Tape) 2-228 to 2-229
 - PO (Power Off Pack) 2-239
 - RP (Ready and Purge Magnetic Tape) 2-273 to 2-274
 - RW (Rewind and Unload Tape) 2-278
 - RY (Ready Peripheral Device) 2-281 to 2-282
 - SN (Purge And Place Tape Number In Label) 2-293 to 2-294
 - SV (Save a Peripheral Unit) 2-309
 - TM (Write Tape Mark) 2-317 to 2-318
 - UL (Assign Unlabeled File) 2-323 to 2-324
- peripherals
 - ALTER (Change System Parameters) 2-16 to 2-20
 - CK (Test Peripheral Device) 2-54
 - COPY (Invoke File Transfer Utility) 2-63 to 2-72
 - COPYADD (Transfer Files) 2-73
 - DA,DAP (Deactivate Pseudo Reader) 2-76 to 2-77
 - DIR (List File Directory) 2-99 to 2-100
 - DISPLAY/WHATS (Display Information) 2-101 to 2-104
 - DL (Delete Peripheral Devices) 2-105
 - FILE (Equate File-Name With File-Identifier Or Medium) 2-132 to 2-140
 - FM (Direct File To Device — Special Forms Response) 2-141
 - FP (Flush Program) 2-143
 - IL (Assign Labeled File) 2-155 to 2-156
 - LD LDP (Create Pseudo Card Deck) 2-175
 - LI (Log In) 2-179
 - LO (Log Out) 2-182
 - MERGE (Transfer Files From Multiple Sources, Resolving Duplicates) 2-189
 - MOVE (Move Files To New Media) 2-190
 - MR (Remove Duplicate File) 2-191 to 2-192
 - OL (Display Peripheral Status) 2-203 to 2-208
 - OU (Direct File To Device) 2-210 to 2-211
 - RD,RDP (Remove Pseudo Decks) 2-260 to 2-261
 - REMOVE (Delete Files) 2-262 to 2-264
 - SHOW (Display System Information) 2-286 to 2-289
 - SI (Set Remote Device Identification) 2-290
 - UNIT (Add Peripheral Devices) 2-325 to 2-335
 - UR (Inhibit or Uninhibit Unit) 2-338 to 2-339
- Peripheral Switch II Glossary-16
- PG (Purge Magnetic Tape) 2-228 to 2-229
- phase encoded (PE) Glossary-16
- phase-encoded magnetic tape (MPE) Glossary-16
- physical subsystem Glossary-17
- physical tape number (PTN) Glossary-17
- PM (Print Memory Dump) 2-230 to 2-238
- PO (Power Off Pack) 2-239
- pool size Glossary-17
- port Glossary-17
- PP (Print Pack Sectors) 2-240
- PR (Change Priority) 2-241 to 2-242
- PRINT (Invokes PBDPRN) 2-243 to 2-250
- printer backup file Glossary-17
- printer files Glossary-17

V Series MCP/VS System Software Operations Guide, Volume 2

Index

printing

KP (Print Disk Segments) 2-165
PB (Print Backup File) 2-220
PM (Print Memory Dump) 2-230 to 2-238
PP (Print Pack Sectors) 2-240
PRINT (Invokes PBDPRN) 2-243 to 2-250
SK (Skip Backup Records) 2-291

priority Glossary-17

PRIORITY (Common Priority Setting) 2-251 to 2-252

privileged mode Glossary-17

PRM (Change Memory Priority) 2-241 to 2-242

PRM (Set Memory Priority) 2-251 to 2-252

PRN option 2-297

processor Glossary-17

processor name Glossary-17

processor number Glossary-17

processor time Glossary-17

Production DBP Glossary-17

program Glossary-17

AX (Pre-Respond To An ACCEPT) 2-25 to 2-26
AX (Respond To An ACCEPT) 2-24
COMPILE (Compile Program) 2-57 to 2-62
DC (Display Compilation Data) 2-89 to 2-90
DISPLAY/WHATS (Display Information) 2-101 to 2-104
DS (Discontinue Program) 2-115 to 2-116
EXECUTE (Execute Library Program) 2-127 to 2-130
FI (List Pack Files In Use) 2-131
FN (List File Names) 2-142
FP (Flush Program) 2-143
GO (Restart A Stopped Program) 2-147
IN (Insert Data Into Program) 2-157 to 2-158
INSERT (Insert Data Into Program) 2-159 to 2-160
LOCK (Lock Function) 2-184
LP (Lock Program) 2-185
MEMORY (Allocate Memory For Program Execution) 2-187 to 2-188
OF (Optional File Indication) 2-200
OK (Continue Processing Suspended Program) 2-201 to 2-2-2
OT (Display Program Data) 2-209
ST (Suspend Program Processing) 2-306
START (Restart A Stopped Program) 2-307
STOP (Suspend Program Processing) 2-308
SW (Set Programmatic Switches) 2-310 to 2-311
UP (Unlock Program) 2-337
VALUE (Insert Value Into A Program) 2-342 to 2-343

programmatic Glossary-17

programmatic switch Glossary-17

program name Glossary-17

PROPS, entering commands through 1-7

PRP (Change Processing And I/O Priority) 2-241 to 2-242

PRP (Set Processing And I/O Priority) 2-251 to 2-252

PRS (Change Scheduling Priority) 2-241 to 2-242

PRS (Set Schedule Priority) 2-251 to 2-252

PS (Analyze Pack Space) 2-253

PSE Glossary-17

pseudo card

BCL (Mark Beginning Of Card Data File) 2-27
CD,CDP (List Inactive Pseudo Decks) 2-47 to 2-48
CV, CVP (Convert Punch Backup File To Pseudo Card Deck) 2-74 to 2-75
DA,DAP (Deactivate Pseudo Reader) 2-76 to 2-77
EBDIC (Mark Beginning Of Card Date File) 2-119
END/ENDCTL (End of Card File or Control Instructions) 2-121
LD, LDP (Create Pseudo Card Deck) 2-175
RD,RDP (Remove Pseudo Decks) 2-260 to 2-261
RN, RNP (Activate Pseudo Reader) 2-270 to 2-271
SD (Set Deck Limit) 2-285

pseudo card deck Glossary-17

PSN Glossary-17

Public Data Network (PDN) Glossary-17

punch backup file Glossary-18

punch file Glossary-18

Q

QD (Query Debug) 2-254

queue Glossary-18

QT (Quit Program Operation) 2-255 to 2-256

QWIK disk Glossary-18

QWIK memory Glossary-18

QWIK option 2-297

QWIK pool Glossary-18

QWIKPOOL (Set Program Overlay Buffer Pool) 2-257

QWKMEM (Set MCP Overlay Buffer Pool) 2-258

R

RA (Remove After-Linkage) 2-259

railroad syntax diagram A-1 to A-6, Glossary-18

constants A-2

explanation of A-1

items of A-2

loops and bridges A-4

optional items A-3

punctuation A-6

required items A-3

variables A-2

RAM Glossary-18

random access memory (RAM) Glossary-18

RD,RDP (Remove Pseudo Decks) 2-260 to 2-261

reader sorters, adding 2-335

Ready Glossary-18

record Glossary-18

release Glossary-18

release level Glossary-18

Remote Job Entry (RJE) Glossary-18. *See* RJE.

remote support Glossary-18

REMOVE (Delete Files) 2-262 to 2-264

RERUN (Log a Rerun Operation) 2-265

V Series MCP/VS System Software Operations Guide, Volume 2

Index

- reset Glossary-18
- restricted master pack Glossary-18
- restricted pack Glossary-18
- restrictions
 - access level 1-12
 - commands invalid from program for SPOM BCT 1-12
 - commands invalid from ZIP or ZIPSPO 1-12
 - commands invalid through card reader 1-12
 - commands invalid through RJE 1-12
 - commands valid from mix number 0 1-12
 - commands valid through WFL 1-12
 - system commands 1-12
- result descriptor (R/D) Glossary-18
- RF, RFP (Remove Backup File From Disk or Pack) 2-266 to 2-267
- RJE Glossary-18
 - commands invalid through 1-12
 - device, adding an 2-334
 - entering commands through 1-7
 - option 2-297
- RJE handler Glossary-18
- RM (Remove Duplicate File) 2-268 to 2-269
- RMOV option 2-297
- RN, RNP (Activate Pseudo Reader) 2-270 to 2-271
- RO (Reset System Options) 2-272
- rollout Glossary-18
- RP (Ready and Purge Magnetic Tape) 2-273 to 2-274
- RPG Glossary-19
- RQ (Remove Storage Queue Entries) 2-275
- RS (Remove Job From Schedule) 2-276
- RTSL option 2-297
- RUN (Initiate Execution Of Program) 2-277
- run log Glossary-19
- RW (Rewind and Unload Tape) 2-278
- RX (Return Deleted Space) 2-279 to 2-280
- RY (Ready Peripheral Device) 2-281 to 2-282
- S
 - S4A, adding an 2-335
 - S4B, adding an 2-335
 - Saved Glossary-19
 - SB (Interrogate Database Activity) 2-283 to 2-284
 - schedule Glossary-19
 - PRS (Change Scheduling Priority) 2-241 to 2-242
 - PRS (Set Schedule Priority) 2-251 to 2-252
 - SCHM option 2-297
 - SD (Set Deck Limit) 2-285
 - sector Glossary-19
 - sector address Glossary-19
 - security
 - BEGINUSER (Mark Beginning Of Job Stream) 2-28 to 2-29
 - BO (Blackout Access Code) 2-39
 - CHARGE (Insert Charge Number) 2-52
 - ENDUSER (Mark End Of Job Stream) 2-123 to 2-124
 - FILE (Equate File-Name With File-Identifier Or Medium) 2-132 to 2-140
 - LI (Log In) 2-179
 - LO (Log Out) 2-182
 - USER (Assign Access Code) 2-340 to 2-341
 - segment Glossary-19
 - SENSITIVEDATA Glossary-19
 - sequential organization Glossary-19
 - serial transmission Glossary-19
 - set Glossary-19
 - shared file I/O operations Glossary-19
 - shared pack Glossary-19
 - shared system
 - DBCA (Clear SSP Address Contention) 2-80 to 2-81
 - DBSR (Produce Block Lockout Table Report) 2-86
 - DBUA (Unlock Address) 2-87
 - DBUL (Clear Locked Files on Shared Systems) 2-88
 - disk only Glossary-19
 - disk and pack Glossary-19
 - initialization records Glossary-19
 - pack Glossary-19
 - SHOW (Display System Information) 2-286 to 2-289
- Shared System Processor (SSP) Glossary-19
- SHOW (Display System Information) 2-286 to 2-289
- SI (Set Remote Device Identification) 2-290
- Signed numeric Glossary-19
- SK (Skip Backup Records) 2-291
- slash parameters 2-61, 2-87
- SM (Send System Message) 2-292
- SN Glossary-19
- SN (Purge And Place Tape Number In Label) 2-293 to 2-294
- snap Glossary-19
- SNAP option 2-297
- snapshot Glossary-20
- SO (Set System Options) 2-295 to 2-298
- soft device Glossary-20
- SP (Display/Set/Change Database Parameters) 2-299
- SPO Glossary-20
- SPO (Simulate SPO Through Card Reader) 2-300
- SPOM Glossary-20
- SPOMESSAGE BCT Glossary-20
 - commands invalid from 1-12
 - entering commands through a 1-6
- SQ (Squash Disk) 2-301
- SQP (Squash Pack) 2-302 to 2-304
- SQUASH 2-302 to 2-304
- SS (Send Message) 2-305
- SSP Glossary-20
 - DBCA (Clear SSP Address Contention) 2-80 to 2-81
 - DBDR (Produce SSP Content Report) 2-83
 - DBUA (Unlock Address) 2-87
 - DBUL (Clear Locked Files on Shared Systems) 2-88
- SSP contention Glossary-20
- ST (Suspend Program Processing) 2-306

V Series MCP/VS System Software Operations Guide, Volume 2

Index

- stacked deck Glossary-20
- stalemate Glossary-20
- START (Restart A Stopped Program) 2-307
- start bit Glossary-20
- start/stop transmission Glossary-20
- station Glossary-20
- statistics Glossary-20
- status Glossary-20
- STGO option 2-297
- STOP (Suspend Program Processing) 2-308
- stop bit Glossary-20
- STOQ Glossary-20
- stoque Glossary-20
 - RQ (Remove Storage Queue Entries) 2-275
 - WQ (Display STOQUE Count) 2-354 to 2-355
- storage queue Glossary-20
- string Glossary-20
- Supervisory Printer Operator Display Terminal (SPO) Glossary-20
- SV (Save a Peripheral Unit) 2-309
- SW (Set Programmatic Switches) 2-310 to 2-311
- switch Glossary-20
- synchronous Glossary-21
- syncpoint Glossary-21
- SYST option 2-297
- system commands 2-1 to 2-365
 - access level restrictions 1-12
 - entering through CANDE 1-7
 - entering through punched cards 1-6
 - entering through PROPS 1-7
 - entering through RJE 1-7
 - entering through SPOMESSAGE BCT 1-6
 - entering through WFL 1-7
 - entering through ZIP 1-5
 - entering through ZIPSPO 1-5
 - general comments 1-8
 - introduction 1-1
 - restrictions 1-12
 - through the ODT 1-3
 - ways to enter 1-1
- system configuration file Glossary-21
- system disk Glossary-21
- system initialization Glossary-21
- system maintenance
 - ALLOCATE (Create Disk Space And Directory Entries) 2-13 to 2-15
 - ALTER (Change System Parameters) 2-16 to 2-20
 - BT (Begin TRAK) 2-40 to 2-41
 - DBCA (Clear SSP Address Contention) 2-80 to 2-81
 - DBCL (Cancel DLP) 2-82
 - DBDR (Produce SSP Content Report) 2-83
 - DBIC (Initialize Controller Parameters) 2-84
 - DBMM (Set/Reset Maintenance Mode) 2-85
 - DBSR (Produce Block Lockout Table Report) 2-86
 - DBUA (Unlock Address) 2-87
 - DBUL (Clear Locked Files on Shared Systems) 2-88
 - DD (Delete DLP) 2-91
 - DISK (Add A Disk) 2-96 to 2-98
 - DL (Delete Peripheral Devices) 2-105
 - DLP (Add DLPs) 2-106 to 2-109
 - DR (Date Reset) 2-114
 - DX (Display DLPs Attached To An Exchange) 2-118
 - ET (End TRAK) 2-126
 - HL (Halt/Load Request) 2-150 to 2-151
 - ID (Invoke DEBUG Session) 2-154
 - IR (Initiate DMSII Recovery) 2-161
 - LH (Load Firmware Into A Channel Controller or DCP) 2-176 to 2-178
 - LN (Transfer and Print Log) 2-180 to 2-181
 - NET (Network Control Command) 2-194 to 2-196
 - NW (Networking Command Prefix) 2-199
 - PACK (Add A Pack) 2-213 to 2-214
 - QD (Query Debug) 2-254
 - QWIKPOOL (Set Program Overlay Buffer Pool) 2-257
 - QWKMEM (Set MCP Overlay Buffer Pool) 2-258
 - RO (Reset System Options) 2-272
 - RQ (Remove Storage Queue Entries) 2-275
 - RX (Return Deleted Space) 2-279 to 2-280
 - SD (Set Deck Limit) 2-285
 - SI (Set Remote Device Identification) 2-290
 - SO (Set System Options) 2-295 to 2-298
 - SP (Display/Set/Change Database Parameters) 2-299
 - SQ (Squash Disk) 2-301
 - SQP (Squash Pack) 2-302 to 2-304
 - TL (Transfer Log) 2-315
 - TR (Time Reset) 2-320
 - UNIT (Add Peripheral Devices) 2-325 to 2-335
 - UR (Inhibit or Uninhibit Unit) 2-338 to 2-339
 - WX (Display Removed Space) 2-358
 - XA (Add/Remove A Channel From An Exchange) 2-361
 - XC (Inhibit/Uninhibit Or Display Status Of A Channel) 2-362
 - XD (Remove Disk Segments) 2-363
 - XM (Remove Memory From System) 2-364
 - XP (Remove Pack Sectors) 2-365
- system security Glossary-21
- system status tables Glossary-21
 - AD (Specify ODT/OCS Screen Parameters) 2-4 to 2-8
- system tape Glossary-21
- SYSTEM/COPY B-1
- SYUP option 2-297
- T
- TABS II Glossary-21
- tailored I/O Glossary-21
- tape
 - AC (Purge And Place Tape Number In Label) 2-2 to 2-3
 - ADD (Copy Files From Library Tape To Disk or Pack) 2-9

V Series MCP/VS System Software Operations Guide, Volume 2

Index

tape (Continued)

- adding tape drives 2-330
- CN (Display Tape Number) 2-55
- FR (Designate Final Reel) 2-144
- LABEL1 (Change Tape Label Format) 2-170 to 2-173
- PG (Purge Magnetic Tape) 2-228 to 2-229
- RP (Ready and Purge Magnetic Tape) 2-273 to 2-274
- RW (Rewind and Unload Tape) 2-278
- SN (Purge And Place Tape Number In Label) 2-293 to 2-294
- TM (Write Tape Mark) 2-317 to 2-318
- UL (Assign Unlabeled File) 2-323 to 2-324

tape mark Glossary-21

TC5 device, adding a 2-334

TELCOM DLP Glossary-21

termination Glossary-21

TERM option 2-297

TEST (Mark A Test Operation) 2-312

Test Bus Interface Glossary-21

test routines Glossary-21

TI (Display Time Statistics) 2-313

TIME (Discontinue Program After a Specified Amount of Time) 2-314

Time Analysis and Billing System II (TABS II) Glossary-21

TIME option 2-297

timesharing Glossary-21

- RUN (Initiate Execution Of Program) 2-277

TL (Transfer Log) 2-315 to 2-316

TM (Write Tape Mark) 2-317 to 2-318

TO (Type System Options) 2-319

TPLB option 2-297

TPMK option 2-297

TR (Time Reset) 2-230

TRACE (Trace a Program) 2-321 to 2-322

trace Glossary-21

- GT (Initiate Trace) 2-148 to 2-149
- NT (Terminate Trace) 2-198

train printer Glossary-21

- adding a 2-333

Train Printer Data Link Processor (TP-DLP) Glossary-21

TRAK Glossary-22

TRAK option 2-297

TRAK utility 2-40 to 2-41, 2-126

transfer rate Glossary-22

translate table file Glossary-22

translation table Glossary-22

TRMD option 2-297

TWA Glossary-22

two-way alternate Glossary-22

two-way simultaneous Glossary-22

TWS Glossary-22

TWX device, adding a 2-334

U

UA Glossary-22

UL (Assign Unlabeled File) 2-323 to 2-324

UN Glossary-22

undigit Glossary-22

Uniline Data Link Processor (DLP)

Uniline firmware Glossary-22

UNIT (Add Peripheral Devices) 2-325 to 2-335

unit Glossary-22

UNLOAD (Move Library Files Between Media, Remove Old File) 2-336, B-8 to B-9

UNLOCK Glossary-22

unrestricted pack Glossary-22

Unsigned numeric Glossary-22

UP (Unlock Program) 2-337

UR (Inhibit or Uninhibit Unit) 2-338 to 2-339

USE records 2-295 to 2-298

USER (Assign Access Code) 2-340 to 2-341

user code Glossary-22

USERHO Glossary-22

user mode Glossary-22

user overlay, QWIKPOOL (Set Program Overlay Buffer Pool) 2-257

user program Glossary-22

USER QWIK memory Glossary-23

- WO (Display Active USER QWIK Operation Counts) 2-352 to 2-353

utility Glossary-23

V

VALUE (Insert Value Into A Program) 2-342 to 2-343

VDD device, adding a 2-334

version date Glossary-23

virtual circuit Glossary-23

Virtual Static Imaging Device (VSID) Glossary-23

volume Glossary-23

VSID Glossary-23

W

waiting Glossary-23

WB (Display Bound Programs) 2-344

WC (Display Available Memory) 2-345

WD (Display Date) 2-346 to 2-347

WFL Glossary-23

- commands invalid through 1-12
- entering commands through 1-7
- QT (Quit Program Operation) 2-255 to 2-256

WJ (Waiting Jobs) 2-348 to 2-349

WM (Display MCP Version) 2-350 to 2-351

WO (Display Active USER QWIK Operation Counts) 2-352 to 2-353

Work Flow Language (WFL) Glossary-23

Working Available Table Glossary-23

WQ (Display STOQUE Count) 2-354 to 2-355

writeahead Glossary-23

WRKP option 2-297

WS (Display Jobs in Schedule) 2-356

WT (Display Time) 2-357

V Series MCP/VS System Software Operations Guide, Volume 2

Index

WX (Display Removed Space) 2-358

WY (Display Job Status) 2-359 to 2-360

X

X.25 Glossary-23

XA (Add/Remove A Channel From An Exchange) 2-361

XC (Inhibit/Uninhibit Or Display Status Of A Channel) 2-362

XD (Remove Disk Segments) 2-363

XM (Remove Memory From System) 2-364

XP (Remove Pack Sectors) 2-365

Z

ZIP Glossary-23

commands invalid from 1-12

entering commands through 1-5

ZIPM option 2-297

ZIPSPO Glossary-23

commands invalid from 1-12

entering commands through a 1-5